

ESTUDO E ANÁLISE DE SOLUÇÕES DE ACESSOS WIMAX PARTILHADOS EM CENÁRIOS RESIDENCIAIS

Jorge Miguel Gil Moreira



Mestrado em Engenharia Electrotécnica e de Computadores

Área de Especialização de Telecomunicações

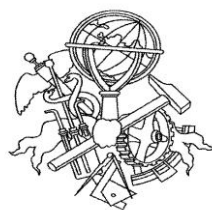
Departamento de Engenharia Electrotécnica

Instituto Superior de Engenharia do Porto

2010

Este relatório satisfaz, parcialmente, os requisitos que constam da Ficha de Disciplina de
Tese/Dissertação, do 2º ano, do Mestrado em Engenharia Electrotécnica e de
Computadores

Candidato: Jorge Miguel Gil Moreira, N° 1030098, 1030098@isep.ipp.pt
Orientação científica: Prof. Doutor Jorge Botelho Mamede, jbm@isep.ipp.pt



Mestrado em Engenharia Electrotécnica e de Computadores
Área de Especialização de Telecomunicações
Departamento de Engenharia Electrotécnica
Instituto Superior de Engenharia do Porto
5 de Dezembro de 2010

Agradecimentos

Agradeço aos meus pais pelos valores que me inculcaram e por todo o apoio que sempre me têm dado.

À minha esposa por toda a motivação, carinho e compreensão.

Por fim, agradeço aos meus Professores e em especial ao orientador da Tese, o Prof. Doutor Jorge Mamede por toda a sua disponibilidade mas também pelos conhecimentos e motivação transmitidos.

Jorge Moreira

Resumo

Esta tese foi desenvolvida na sequência do crescente interesse pela tecnologia WIMAX (*Worldwide Interoperability for Microwave Access*) como meio de acesso banda larga à rede Internet. O trabalho desenvolvido vem contribuir com a aplicação SimWimax para a simulação de cenários residenciais com acesso WIMAX.

O simulador utilizado neste trabalho é o NS-2, ferramenta muito usada em trabalhos de simulação de redes computacionais com ou sem fios. No entanto, WIMAX não está incluído no NS-2, bem como outros módulos para simulação de tráfegos. Vários investigadores disponibilizam diferentes módulos WIMAX para este simulador, sendo adoptado o módulo da UNICAMP. Para simular diversos tipos de tráfego HTTP, VOIP e *Video on Demand* foram adoptados trabalhos de outros investigadores.

Com este conjunto de ferramentas foi construída uma aplicação Web para manipulação das simulações e para análise dos resultados.

O SimWimax permite não só configurar e simular, mas também fazer todo o tratamento de dados estatísticos em termos de taxa de transferência e atraso, apresentando-os no final da simulação sob a forma de gráficos. Esta aplicação torna-se também uma ferramenta de estudo e projecto, permitindo a um utilizador sem conhecimentos específicos de simulação com o NS-2 fazer um estudo de determinados cenários de aplicação da tecnologia WIMAX.

Palavras-Chave

SimWimax, WIMAX, simulação, NS-2, interface web.

Abstract

This thesis has been developed due to the growing interest in the WIMAX (*Worldwide Interoperability for Microwave Access*) technology as a way to access Internet. The work developed has contributed with a SimWimax web software to the residential scenarios' simulation with WIMAX access technology.

The simulator used in this project is NS-2, a widely used tool in wired or wireless network simulations. However, WIMAX is not included in NS-2, as well as other traffic simulation modules. Several investigators have developed different WIMAX modules to this simulator but the chosen module was UNICAMP. To simulate HTTP, VOIP and *Video on Demand* traffic the work of other investigators was adopted.

With this set of tools a web application was developed to simulate and to analyse the results.

SimWimax is able to setup, simulate and process data in terms of throughput and delays. This data is shown in form of graphics. The application is also a study tool since a user without NS-2 knowledge can study the different scenarios using WIMAX access.

Keywords

SimWimax, WIMAX, simulation, NS-2, web interface.

ÍNDICE

AGRADECIMENTOS	V
RESUMO.....	VII
ABSTRACT.....	IX
ÍNDICE DE FIGURAS	XIII
ÍNDICE DE TABELAS.....	XV
ACRÓNIMOS	XVII
1. INTRODUÇÃO.....	1
1.1. CONTEXTUALIZAÇÃO	1
1.2. OBJECTIVOS.....	1
1.3. ORGANIZAÇÃO DO RELATÓRIO	2
2. WIMAX.....	3
3. SIMULAÇÃO	17
3.1. SIMULADOR NS-2.....	17
3.2. SIMULAÇÃO WIMAX – MODELOS ADOPTADOS	26
4. SIMWIMAX - APLICAÇÃO DE SIMULAÇÃO WIMAX	33
4.1. APRESENTAÇÃO GENÉRICA DA APLICAÇÃO	33
4.2. CENÁRIOS DE APLICAÇÃO RESIDENCIAL.....	35
4.3. PERFIS DE UTILIZADOR	36
4.4. FRONTEND GRÁFICO	38
4.5. PROCESSO DE SIMULAÇÃO	50
5. ANÁLISE DE RESULTADOS	53
6. CONCLUSÕES.....	65
6.1. DESENVOLVIMENTOS FUTUROS.....	66
REFERÊNCIAS DOCUMENTAIS	67

Índice de Figuras

Figura 1	Dimensão Redes [40].....	4
Figura 2	Configuração Ponto-a-Ponto [41].....	6
Figura 3	Configuração Mesh [41]	6
Figura 4	Configuração Ponto-Multiponto [41].....	7
Figura 5	Camada MAC em WIMAX	10
Figura 6	Blocos importantes do Simulador NS-2[7]	18
Figura 7	Escalonador de Eventos [39]	19
Figura 8	Arquitectura NS-2 [39]	20
Figura 9	Tipos de nós <i>unicast</i> e <i>multicast</i> [39]	21
Figura 10	Ligação unidireccional entre nós [7].....	21
Figura 11	Ligação com recolha de dados [7]	22
Figura 12	Exemplo Trace gerado pelo NS-2 [39]	22
Figura 13	<i>Streams</i> e <i>substreams</i> do gerador de números aleatórios [7].....	25
Figura 14	Modelo esquemático do módulo WIMAX [12].....	27
Figura 15	Blocos funcionais do módulo VOIP [15].....	28
Figura 16	Modelo para produzir ficheiros <i>Trace</i> de Video [18]	30
Figura 17	Características de cada sequência de vídeo [13].....	30
Figura 18	Blocos da aplicação SimWimax	34
Figura 19	Arquitectura do Cenário de aglomerado de habitações.....	35
Figura 20	Arquitectura do Cenário de edifício comum	36
Figura 21	Páginas <i>web</i> da aplicação SimWimax.....	38
Figura 22	Menu principal SimWimax	39
Figura 23	Página de Configuração Básica SimWimax.....	39
Figura 24	Página de Configuração Avançada SimWimax.....	40
Figura 25	Opção de configuração módulo VoD.....	40
Figura 26	Opções de configuração módulo VOIP.....	40
Figura 27	Opções de configuração módulo HTTP.....	41
Figura 28	Opções configuração geral SimWimax.....	41
Figura 29	Opções de configuração avançada no SimWimax.....	41
Figura 30	Página de simulação Cenário Residencial - Aglomerado de habitações	42
Figura 31	Parâmetros de configuração Cenário Aglomerado de habitações.....	42
Figura 32	Exemplo da página de resultados do SimWimax	43
Figura 33	Processo output SimWimax	44
Figura 34	Configuração codec na aplicação SimWimax	44
Figura 35	<i>Parsing</i> do ficheiro <i>trace</i>	48

Figura 36	<i>Parsing</i> dos vários ficheiros trace.....	48
Figura 37	Cenário de teste 1	54
Figura 38	Taxa de transferência e atraso tráfego VoD Cenário 1	55
Figura 39	Taxa de transferência e atraso VOIP Cenário 1.....	55
Figura 40	Cenário para teste 1.1.....	56
Figura 41	Tráfego HTTP com distribuição HTTP-1	57
Figura 42	Cenário de teste 1.3.....	58
Figura 43	Taxa de transferência e Atraso VOIP cenário 1.3	58
Figura 44	Cenário de teste 2	59
Figura 45	Taxa de transferência VoD Cenário 2.....	60
Figura 46	Taxa de transferência VOIP Cenário 2	60
Figura 47	Taxa de transferência e atraso HTTP Cenário 2.....	61
Figura 48	Cenário de teste 3	61
Figura 49	Taxa de transferência e atraso VoD Cenário 3	62
Figura 50	Taxa de transferência VoD Cenário 3.....	63
Figura 51	Taxa de transferência e atraso HTTP Cenário 3.....	63

Índice de Tabelas

Tabela 1	Estudo comparativo de módulos WIMAX para NS-2 [8]	26
Tabela 2	Distribuições do parâmetro \bar{A}	32
Tabela 3	Parâmetros Cenário Teste 1	54
Tabela 4	Parâmetros para Cenário teste 1.1	57
Tabela 5	Parâmetros para Cenário de teste 1.3	58
Tabela 6	Parâmetros para Cenário de teste 2	60
Tabela 7	Parâmetros Cenário de teste 3	62

Acrónimos

ARQ	– Automatic Repeat Request
BS	– Base Station
CRM	– Customer Relationship Management
DSL	– Digital Subscriber Line
ertPS	– Extended real time Polling Service
ETSI	– European Telecommunications Standarts Institute
FBSS	– Fast Base Station Switching
FDD	– Frequency Division Duplex
HHO	– Hard Handover
HTTP	– Hypertext Transfer Protocol
IEEE	– Institute of Electrical and Electronics Engineers
ISP	– Internet Service Provider
LAN	– Local Area Network
LOS	– Line of Sight
MAC	– Medium Access Control
MAN	– Metropolitan Area Network
LAN	– Local Area Network

MDHO	– Macro Diversity Handover
MPEG	– Moving Picture Experts Group
MSDU	– MAC Service Data Unit
NDSL	– Networks & Distributed Systems Laboratory
NIST	– National Institute of Standards and Technology
NLOS	– Non Line of Sight
NS	– Network Simulator
OFDM	– Orthogonal Frequency Division Multiplexing
OFDMA	– Orthogonal Frequency Division Multiple Access
PHP	– Hypertext Preprocessor
PHY	– Physical
PKM	– Key Management Protocol
PMP	– Point to Multipoint
PTP	– Point to Point
rtPS	– Real Time Polling Service
nrtPS	– Non Real Time Pooling Service
SA	– Security Association
SC	– Single Carrier
SDU	– Service Data Unit

TCL	– Tool Comment Language
TDD	– Time Division Duplex
UNICAMP	– Universidade Estadual de Campinas
VOIP	– Voice over IP
WI-FI	– Wireless Fidelity
WIMAX	– Worldwide Interoperability for Microwave Access
Wi-Fi	– Wireless Fidelity
WirelessMAN	– Wireless Metropolitan Area Network
WirelessHUMAN	– Wireless High-Speed Unlicensed Metropolitan Area Network
WLAN	– Wireless Local Area Network

1. INTRODUÇÃO

1.1. CONTEXTUALIZAÇÃO

Actualmente, o estudo de tecnologias de acesso de banda larga WIMAX tem vindo a crescer, devido ao conjunto de vantagens que este meio de acesso apresenta, tais como: altos débitos, escalabilidade e baixos custos de implementação. Isto tem motivado tanto fabricantes como investigadores a investirem nesta tecnologia.

O WIMAX foi normalizado pelo *Institute of Electrical and Electronics Engineers* (IEEE) no standard IEEE 802.16. Esta norma foi implementada com o intuito de regular mecanismos de acesso ao meio rádio e de Qualidade de Serviço (QoS). Este último aspecto foi deixado em aberto, cabendo aos fabricantes desenvolver os seus próprios algoritmos diferenciando assim os seus produtos. Os mecanismos de QoS são essenciais neste tipo de acesso, podendo conferir ao sistema mais fiabilidade e uma melhor gestão dos recursos.

Este trabalho surge como mais uma contribuição para o estudo e implementação do WIMAX focalizado no acesso *wireless* fixo num cenário residencial.

1.2. OBJECTIVOS

O objectivo deste trabalho é fazer um estudo e análise de cenários residenciais tendo como tecnologia de acesso o WIMAX.

Pretende-se avaliar a viabilidade deste tipo de acesso *wireless* fixo para este tipo de utilizadores residenciais visto terem diferentes modos de utilização dos recursos que podem ter diversos efeitos no dimensionamento do sistema de acesso.

Para permitir e facilitar estudo será desenvolvida uma ferramenta específica com interface Web, o SimWimax.

A aplicação SimWimax permitirá realizar de uma forma mais rápida a simulação e parametrização de cenários de utilização da tecnologia WIMAX para servir conjuntos de utilizadores. Adicionalmente, possibilitará também analisar os resultados obtidos sob a forma de gráficos.

1.3. ORGANIZAÇÃO DO RELATÓRIO

Este documento encontra-se estruturado em seis capítulos. O primeiro capítulo pretende enquadrar a tese e contextualizar o problema. O segundo descreve a tecnologia de acesso WIMAX utilizada. No terceiro refere-se o simulador usado bem como os modelos de tráfego considerados. O quarto capítulo descreve a aplicação SimWimax desenvolvida. No quinto capítulo fazem-se testes aos modelos adoptados. No capítulo final são tiradas conclusões em relação aos testes efectuados e são mencionados possíveis desenvolvimentos tanto a nível de módulos como da aplicação SimWimax.

2. WIMAX

Actualmente, a tecnologia de acesso à Internet sem fios que está em destaque é o WIMAX. O WIMAX é uma tecnologia sem fios de banda larga que surge como tecnologia de acesso concorrente ao tradicional DSL e Cabo. Esta tecnologia de acesso apresenta variadas vantagens como: ligações de alto débito, elevada escalabilidade, custos de implementação e manutenção dos equipamento inferiores quando comparado com acessos cablados (como o ADSL ou Cabo).

Oferece um serviço à escala *Metropolitan Area Network* (MAN) podendo atingir um alcance até 50km e uma taxa de transmissão de 70 Mbps. As redes MAN são redes de dados desenvolvidas através de infra-estruturas de alta velocidade, capazes de servir uma área de dimensão equivalente a uma cidade. A Figura 1 ilustra diferentes dimensões de redes existentes.

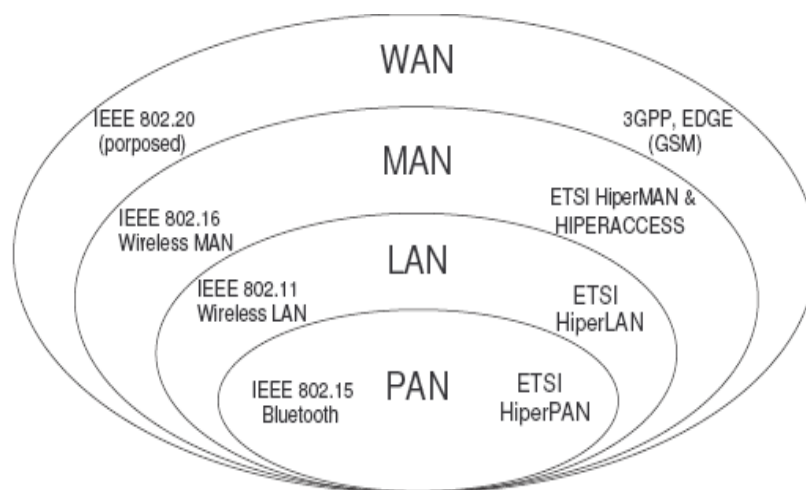


Figura 1 Dimensão Redes [40]

O WIMAX enquadra-se entre a WAN (*World Area Network*), que apresenta uma cobertura global, e a LAN (*Local Area Network*), de menor dimensão, apresentando uma cobertura capaz de servir áreas de pequena dimensão como grupos de residências ou conjunto de prédios.

Associada à LAN está a WLAN (*Wireless LAN*) que é da mesma escala que a LAN mas usa transmissões sem fios via rádio em substituição do cabo. A norma 802.11, também referenciada como WI-FI, está associada à escala WLAN. Efectuando um paralelismo entre WI-FI e WIMAX, é possível afirmar que o WIMAX, padronizado pela norma 802.16, está para a MAN assim como o WI-FI está para a LAN, apresentando velocidades de transmissão muito mais atractivas.

Em 1998, o *Institute of Electrical and Electronic Engineers* (IEEE) formou um grupo chamado 802.16 para desenvolver um standard para o WirelessMAN. Originalmente, este grupo centrou-se no desenvolvimento de uma solução para as bandas 10GHz a 66GHz, com o objectivo de permitir aos operadores oferecer ligações de alta velocidade a clientes empresariais que não pudessem ter acesso fibra. Esta ligação era do tipo ponto-multiponto em linha de vista (LOS). Foi então aprovado um standard WirelessMAN-SC em Dezembro de 2001[4]. Este standard especificava aspectos na camada física (como a utilização de portadora única) e no controlo de acesso ao meio (MAC) [1].

Mais tarde surgiram novos estudos para desenvolver um standard centrado noutras frequências (2GHz a 11GHz) com vista a conferir ao protocolo ligações sem linha de vista (NLOS). O novo standard teve o nome de 802.16a, finalizado em 2003. Outros aspectos na

camada física foram também alterados para suportar NLOS: a camada física passou a suportar a *Orthogonal Frequency Division Multiplexing* - OFDM.

Revisões foram feitas à norma 802.16a e finalizadas em 2004. O novo standard IEEE 802.16-2004 [3] agrupa e substitui o 802.16, 802.16a e 802.16c, passando a ser a base para um novo standard HIPERMAN (*high-performance metropolitan área network*) pelo *European Telecommunications Standards Institute* - ETSI.

Em 2003, o grupo 802.16 iniciou trabalhos para permitir suportar aplicações móveis. Esta revisão 802.16e, terminada em Dezembro de 2005, foi publicada formalmente como IEEE 802.16e-2005. Neste standard são especificados, novamente, modificações a nível de camada física um OFDM escalável e várias alterações também a nível MAC para acomodar mobilidade de alta velocidade.

Inúmeras revisões originaram novos standards: os fabricantes de equipamentos, confrontados com esta situação, não sabiam quais das opções implementar nos seus produtos de maneira que existisse interoperabilidade na família 802.16. Surgiu, então, o WIMAX Forum [2], uma entidade composta por vários sectores tecnológicos como companhias de semicondutores, fabricantes de equipamentos, integradores de sistemas e fornecedores de serviços Internet (ISP), com vista a testar e certificar produtos. O trabalho realizado pelo WIMAX fórum consiste em assegurar a interoperabilidade e conformidade entre os equipamentos de acesso *wireless broadband* baseados nas normas IEEE 802.16. Estas empresas juntam os seus esforços para combinarem as duas normas, criando um processo de certificação para produtos que operam com a tecnologia WIMAX.

2.1.1. APLICAÇÕES

O acesso à tecnologia wireless WIMAX pode ter duas aplicações: acesso fixo e acesso móvel. Aplicações utilizando o acesso fixo podem ser classificadas como ponto-a-ponto, ponto-multiponto ou multiponto a multiponto (mesh). Em todas as topologias existem dois elementos: a Base Station (BS) e a Subscriber Station (SS). A BS é a unidade central da rede que faz toda a gestão da mesma. As SS são estações-cliente da rede espalhadas geograficamente.

A ligação de dois edifícios numa mesma rede de alto débito é um exemplo de uma ligação ponto a ponto.

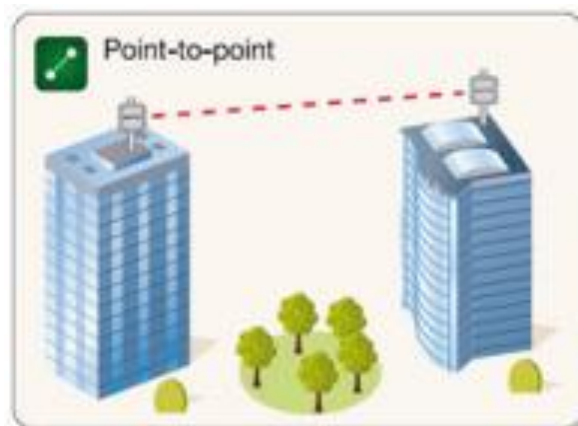


Figura 2 Configuração Ponto-a-Ponto [41]

Ligações em arquitectura em rede ou *mesh*: em sistemas multiponto-a-multiponto ou *mesh*, não existe centralização da BS. Quanto mais SSs entrarem na rede, maior será a área de cobertura. Neste caso, as SSs podem comunicar directamente entre si. A estação que não possuir linha directa de sinal entre a BS e a SS (LOS – “line-of-sight”) consegue receber o tráfego de outra estação através do equipamento de um outro assinante.

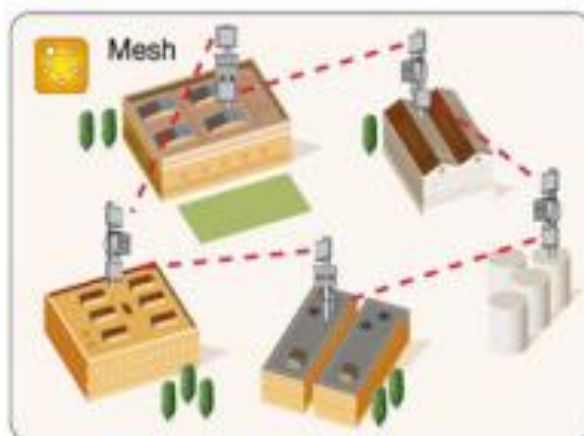


Figura 3 Configuração Mesh [41]

Ligações ponto-multiponto podem ter várias aplicações: acesso banda larga para residências/prédios de habitação ou aglomerado de habitações, pequenas e médias empresas e ligações de alto débito para empresas ou outras entidades com necessidade de ligações com débitos elevados. Nas ligações ponto-multiponto existe uma entidade principal, a BS, e clientes espalhados designados como SS.

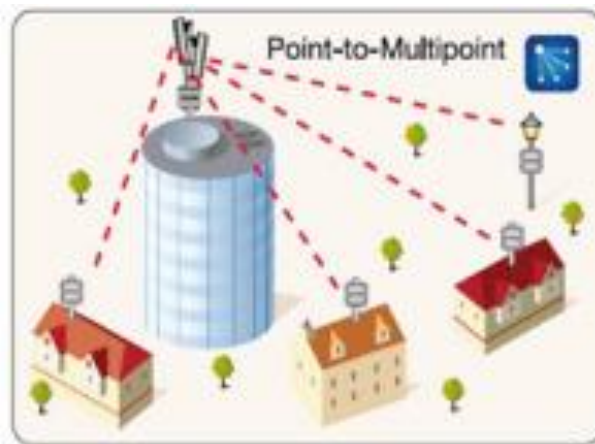


Figura 4 Configuração Ponto-Multiponto [41]

Cenário Residencial – Aglomerado/bairro de Habitações

O cenário residencial (um aglomerado de habitações ou um edifício comum) numa configuração ponto-multiponto é claramente uma das aplicações alvo da tecnologia WIMAX num futuro próximo.

O serviço banda larga utilizando WIMAX fixo inclui acesso Internet de alta velocidade, voz sobre IP - VOIP, Video on Demand. O acesso wireless fixo possui várias vantagens em relação ao acesso cablado, como DSL ou cabo. Estas vantagens incluem: custo de implementação mais baixo, rápida e mais fácil manutenção do sistema [3].

A nível de requisitos de instalação de equipamentos no cliente (na SS), dois modelos de instalação podem ser adoptados: antena interna, à semelhança do habitual modem DSL ou Cabo (neste caso um modem-router rádio) ou antena externa.

Uma vantagem do uso de antena exterior é uma maior capacidade de captação de sinal, melhorando a performance do sistema. Em termos de instalação, já nem todos os clientes finais conseguem ter meios para o fazer (ao contrário da instalação de um modem/router rádio). A instalação de uma antena no exterior terá que ser feita, na maioria dos casos, por pessoal especializado. Além disso, o custo do equipamento e sua instalação poderá ser mais elevado.

Este tipo de acesso fixo wireless está ainda a ser divulgado neste mercado residencial. Será direccionada para regiões remotas ou rurais em que a instalação de rede cablada estruturada seja dispendiosa.

China, Índia, Rússia, Indonésia, América Latina e Europa de Leste possuem rede estruturada cablada insuficiente ou, na maioria, em mau estado mas procuram estar ao nível de outros em termos de acesso e disponibilidade de serviços Internet. Aqui o WIMAX representa a solução mais apropriada em termos de custo de implementação, tempo e da qualidade dos serviços proporcionada [3].

Cenário Residencial – Edifício Comum

Um dos cenários em que o acesso wireless WIMAX é um forte candidato será no caso em que existe um Edifício Comum, podendo assegurar soluções competitivas em ligações de muito alto débito ou ligações dedicadas T1/E1.

Em edifícios habitacionais de grande dimensão, que podem não ter acesso a ligações, por exemplo, em fibra, o WIMAX surge como forte alternativa tecnológica de comunicações.

Outra aplicação desta tecnologia, que tem registado uma crescente procura, é a implementação de hotspots Wi-Fi. Actualmente assegurar serviço Internet a locais como centros de convenções/exposições, hotéis, aeroportos e pequenos negócios de restauração é um requisito do mercado.

2.1.2. ACESSO MÓVEL WIMAX

Para os utilizadores acostumados a ter em casa ou no trabalho acessos internet rápidos é normal que, com o aparecimento massivo de *smartphones* e portáteis, a procura por serviços similares num contexto móvel ou nómada tornou-os imperativos.

Existem vários cenários de mobilidade:

- Nómada: o utilizador liga-se a uma estação fixa e pode ligar-se novamente de um ponto geograficamente diferente;
- Portátil: é um tipo de acesso nómada mas a partir de um dispositivo portátil, como uma placa de rede sem fios.
- Mobilidade Simples: o cliente pode mover-se até 60km/h com breve interrupção de serviço (menos de 1 segundo);
- Mobilidade Total: possibilidade de mobilidade até 120km/h sem interrupção de serviço.

Handover –Métodos para ligação à BS

Na norma IEEE 802.16e-2005 estão previstos três métodos de *handover*, sendo só um deles obrigatório e os restantes opcionais[4].

O primeiro método de *handover*, o obrigatório, tem o nome de *hard handover* (HHO). HHO implica uma transferência abrupta de ligação de uma BS para outra. As decisões de *handoff* são feitas pela BS, MS (Mobile Station), baseado em medidas feitas pela MS. A MS periodicamente faz um varrimento rádio, em instantes definidos pela BS, e mede a qualidade de sinal de estações BS vizinhas. Caso as medições sejam favoráveis à mudança de BS, a MS inicia os procedimentos de associação à nova BS.

Os métodos opcionais de *handoff* suportados pela norma são: *fast base station switching* (FBSS) e *macro density handover* (MDHO). Nestes dois métodos, a MS mantém uma ligação activa com mais do que uma BS. No caso da FBSS, a MS contém uma lista de BS envolvidas – *active set*. A MS monitoriza continuamente a lista de BS envolvidas e mantém uma ligação activa com as BS, mas só comunica com uma só BS, chamada *anchor BS*. Quando existe necessidade de mudar de BS, a ligação é transferida da BS anterior para a nova BS.

O *macro density handover* é similar ao FBSS. Difere num ponto em que a MS comunica no *downlink* e *uplink* com todas as BS presentes na *active set* – designado neste caso por *diversity set*. No *downlink* a MS recebe múltiplas cópias dos dados que são tratados na MS usando técnicas de ordenação e combinação. No *uplink*, em que a MS envia dados para a BS, é escolhido o melhor *link* para fazer upload dos dados.

FBSS e MDHO oferecem performances superiores à HHO mas os requisitos na BS são vários: sincronização entre as BS envolvidas, utilização da mesma frequência e partilha de informação relativa a entradas e saídas das MS na rede.

2.1.3. CAMADA MAC

A camada *Medium Access Control* MAC é responsável por controlar e multiplexar as várias ligações sobre a mesma camada física. Algumas das funções mais importantes desta camada são listadas a seguir:

- Segmentar e concatenar os *Service Data Units* (SDU's) recebidos de camadas superiores em MAC PDU (*Protocol Data Units*), construindo o bloco mais básico do *payload* da camada MAC;
- Seleccionar o perfil de *burst* e nível de energia utilizado para a transmissão os MAC PDU's;
- Retransmissão dos MAC PDU's recebidos erradamente por um receptor quando é utilizado o mecanismo *Automatic Repeat Request* - ARQ;
- Assegurar o controlo de QoS e prioridade no tratamento dos MAC PDU's;
- Controlar o envio dos MAC PDU's para a camada física PHY;
- Suporte às camadas superiores para gestão de mobilidade;
- Gerir aspectos de segurança;
- Promover modos *power-saving* e operação *idle-mode*.

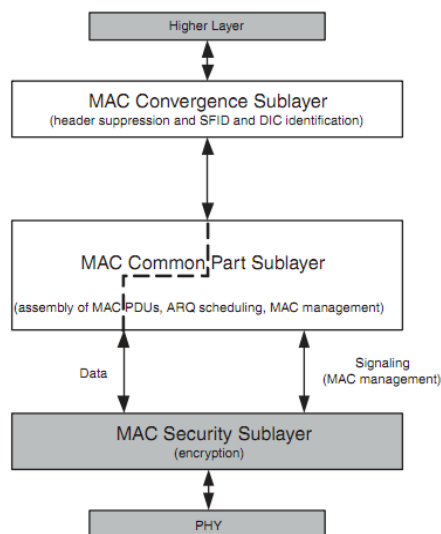


Figura 5 Camada MAC em WIMAX

O detalhe da camada MAC é mostrada na Figura 5 e é dividida em três partes distintas: a *convergence sublayer-CS*, a *sublayer common-part* e *sublayer security*. A CS é uma interface entre a camada MAC e a camada 3 e recebe pacotes de dados da camada superior. Estes pacotes vindos da camada superior têm o nome de *service data units* (SDU).

A CS é responsável por operações que são dependentes da natureza do protocolo da camada superior, como compressão de cabeçalho e atribuição de endereços.

A CS pode ser visto como camada de adaptação que faz a ponte entre a camada superior e seus requisitos com as camadas MAC e PHY da rede WIMAX.

A camada *common-part sublayer* realiza todas as operações a nível de pacote que são: fragmenta e concatena de SDU's para MAC PDU's, transmite MAC PDU's e controla o QoS e ARQ. A camada de segurança envolve responsabilidades a nível de encriptação, autorização e assegura a correcta troca de chaves de encriptação entre a BS e as SS.

Convergence Sublayer

A CS tem a função de mapear os endereços de camadas superiores, como endereços IP ou dos SDU's para a ligação entre MAC e PHY. Esta funcionalidade é necessária porque não existe visibilidade dos endereços por parte das camadas superiores para as camadas MAC e PHY, portanto, a CS trata dessa associação.

A camada MAC é orientada às ligações e identifica a ligação lógica entre BS e SS por um CID – *connection identifier*. Os CID para *downlink* e *uplink* são diferentes. O CID pode ser visto como uma segunda camada temporária e dinâmica da BS para identificar ligações entre a MAC/PHY de um cliente, ligação essa utilizada para transmissão de dados e tráfego de controlo. Para mapear o CID para as camadas superiores a CS precisa de associar o CID e o respectivo endereço de destino. Os SDU podem ser transmitidos por diferentes ligações, visto que a atribuição dos CID pela CS é baseado não só pelo endereço destino mas também por parâmetros QoS, *service flow* ID SFID e endereço origem.

Compressão do cabeçalho do pacote

Outra função importante da CS é a *packet header compression* – PHS. O lado do emissor envolve a remoção de partes repetidas em cada SDU. Por exemplo se os SDU's são entregues em formato pacote IP na CS, o endereço origem e destino IP em cada pacote não muda de um pacote para o seguinte, podendo através deste mecanismo ser removido antes de ser transmitido. De um modo análogo, a par do cabeçalho, pode ser reinserido no SDU antes de ser entregue às camadas superiores. O protocolo PHS estabelece e mantém um determinado grau de sincronismo entre as CS do emissor e receptor.

No WIMAX a implementação do PHS é opcional no entanto é implementado porque garante eficiência em redes com serviços VOIP.

O funcionamento do PHS é baseado numa regra PHS, que fornece os parâmetros relativos à compressão do cabeçalho do SDU. Quando um SDU chega, a CS determina a regra a ser utilizada baseando-se nos endereços origem e destino. Também depende do tipo de serviço como VOIP, Http ou Ftp visto que o número de bytes a serem comprimidos ou compactados dependem do tipo de serviço.

Alocação de largura de banda

No *downlink* as decisões de alocação de largura de banda das várias SS são feitas pela BS baseado no CID. Não requer intervenção das SSs. A BS reserva recursos para o pedido e envia a indicação da alocação para a SS por via de uma mensagem DL-MAP [4].

No *uplink* a SS faz um pedido de largura de banda através de um MAC PDU específico de pedido de banda ou *piggybacking* num MAC PDU genérico.

Os pedidos de banda podem ser incrementais (em que a BS apenas adiciona a percentagem de banda necessária) ou em agregado (em que a BS substitui o pedido de banda por um mais actual).

Este processo de pedido de largura de banda tem o nome de *polling*. Quando uma só SS faz um pedido de largura de banda tem o nome de *unicast*. Assim, são reservados recursos para que esta SS possa fazer pedidos de largura de banda. Basta para isso enviar um PDU a fazer esse pedido. A BS aloca recursos e envia a informação à SS por meio de uma mensagem UL-MAP na subframe de DL-MAP. NA UL-MAP é indicado o *primary* CID da SS para indicar a alocação de banda para essa mesma SS. Este *primary* CID é guardado na SS durante os procedimentos de registo na rede e inicialização e é utilizado para todas as trocas de mensagens de controlo a nível MAC.

A SS pode também requisitar dinamicamente os CID, *secondary* CID, que são utilizados somente para envio de dados.

Caso a largura de banda seja insuficiente para fazer o *polling* individualmente, a cada SS é feito um *polling* em *multicast* ou *broadcast* a todas as SS. Todas as SS podem fazer pedidos de banda durante o *polling*.

Quando a SS necessita de enviar um pedido durante o *polling multicast*, primeiro entra numa fase de contenção, seleccionando um número aleatório baseado numa distribuição uniforme entre valor 0 e um valor de *backoff window*. Este valor aleatório gerado indica o número de oportunidades de transmissão que a SS terá que esperar antes de enviar o pedido de banda. A *backoff window* é o número máximo de oportunidades de transmissão que a SS terá que esperar antes de fazer o pedido de banda. Caso não receba a oportunidade de transmissão dentro do tempo especificado, a SS assume que a oportunidade de transmissão foi perdida devido a colisões, aumenta o *backoff window* e repete o processo. Se a banda mesmo assim não for alocada, o MAC PDU é descartado[3].

Escalonamento

A camada MAC define vários tipos de escalonador para tratar os SDU's e MAC PDU com diferentes requisitos QoS [5].

- *Unsolicited grant service* – UGS- destinado a suportar serviços em tempo real com pacotes de dados de tamanho fixos emitidos em intervalos periódicos de tempo, tal como T1/E1 e VOIP sem omissão de silêncio. Há vários parâmetros a considerar para este serviço: taxa de transferência máxima de tráfego, latência máxima, *jitter* máximo e taxa de transferência mínima.
- *Real-time Polling Service* – rtPS – projectado para sustentar fluxos de dados em tempo real com tamanho variável de pacotes de dados emitidos em intervalos periódicos de tempo, como vídeos MPEG. Parâmetros a ter em conta: taxa de transferência mínima e máxima efectiva de tráfego, máxima latência.
- *Non real time polling services* – nrtPS – similar ao rtPS excepto no caso em que a SS possa utilizar uma política de contenção no *uplink* para requisitar banda. Suporta fluxos de dados com tolerância de atraso, consistindo em pacotes de dados de tamanho variável em que a mínima taxa de tráfego é satisfeita, como o Ftp. Parâmetros importantes: taxa de transferência efectiva mínima, taxa de transferência máxima e prioridades de tráfego.
- *Best Effort* – BE – serviço em que não é necessário assegurar nível mínimo de serviço, logo, pode ser assegurado numa base de espaço/disponibilidade. Parâmetros importantes: taxa de transferência máxima e prioridades de tráfego.

- Extended real-time Polling Service – ertPS – serviço introduzido aquando da nova norma IEEE 802.16e. Projectado para ter eficiência de ambos serviços UGS e rtPS. A BS deve suportar todos os serviços disponíveis pela classe UGS salvaguardando a latência. A diferença é que, no serviço UGS, as alocações são de tamanho fixo enquanto no ertPS as alocações são dinâmicas. A aplicação típica para este tipo de serviço é o VOIP.

Subcamada de Segurança

Esta subcamada fornece privacidade aos assinantes da rede sem fios através da encriptação das ligações entre a SS e a BS, assim como uma forte protecção contra roubo de serviços. A BS protege contra acessos não autorizados aos serviços de transporte de dados forçando a encriptação dos fluxos de serviço através da rede.

Nesta subcamada é empregue um protocolo de encapsulamento, para encriptação dos pacotes de dados (este protocolo também define as criptografias suportadas), análise dos dados encriptados, algoritmos de autenticação e regras de aplicação destes algoritmos no MAC PDU *payload*. Também é utilizado um protocolo de gestão de chaves (PKM – *Key Management Protocol*). Este protocolo é utilizado pela SS para obter autorização e tráfego dos dados da chave da BS, reautorização periódica e actualização de chave. O PKM utiliza certificação digital X.509, algoritmo de encriptação RSA de chave pública (PKCS #1) e fortes algoritmos de encriptação para actuar na troca de chaves entre a SS e a BS. Este protocolo foi criado através do conceito de *security associations* (SAs). As SAs contêm as informações sobre quais os algoritmos de encriptação que devem ser utilizados e qual chave a aplicar [3].

Camada Física PHY

A camada física opera num formato de frames, que são organizadas em intervalos de tempo, chamados slots físicos. Em cada frame existe um subframe *downlink* e um subframe *uplink*. O subframe *downlink* é usado pela BS para a transmissão de dados e de informações de controlo para as SSs. O subframe *uplink* é partilhado por todas as SSs para transmissões que têm como destino a BS. As principais funções da camada física são:

- Transmissão dos MAC PDUs;
- Definição das técnicas de transmissão digital: modulação e codificação;

- Definição do espectro;
- Correção de erro;
- Construção das frames e subframes de transmissão.

A norma IEEE 802.16a/d possui diferentes especificações de interfaces rádio [6]:

- WirelessMAN-SC (*Single Carrier*): a camada física com interface rádio usa uma única portadora que opera em altas frequências, sendo assim necessária a garantia de linha de vista.
- WirelessMAN-SCa (*Single Carrier Adaptive*): usa uma única portadora adaptativa para a interface rádio. Assim, é possível a transmissão fora da linha de vista.
- WirelessMAN-OFDM (*Orthogonal Frequency Division Multiplexing*): multiplexação por divisão de frequência ortogonal com 256 portadoras: 192 são utilizadas para dados, 56 são anuladas, por serem utilizadas como banda de guarda e 8 são pilotos, ou seja, utilizadas para auxiliar na sincronização e igualização. O acesso múltiplo para diferentes estações clientes (SSs) é baseado em divisão de tempo (TDMA - *Time Division Multiple Access*). É a especificação preferida pelo fórum dos fabricantes (WiMAX) por sofrer menos com interferências e permitir o cálculo da transformada rápida de Fourier (FFT - *Fourier Fast Transform*).
- WirelessMAN-OFDMA (*Orthogonal Frequency Division Multiple Access*): OFDM com 2048 portadoras. O acesso múltiplo é fornecido atribuindo um subconjunto de portadoras para um único receptor. Este esquema de acesso múltiplo combina TDMA e OFDMA.
- WirelessMAN HUMAN (*High-speed Unlicensed MAN*): especificação para redes metropolitanas não licenciadas, suporta os mesmos padrões, possuindo alguns componentes específicos de canalização e transmissão de máscara espectral.

3. SIMULAÇÃO

3.1. SIMULADOR NS-2

No estudo de redes, a simulação é uma ferramenta imprescindível para se fazer uma representação o mais aproximada de uma determinada situação real alvo de estudo.

Este estudo recorre então a ferramentas de simulação. A ferramenta escolhida é designada por Network Simulator 2 – NS2 [7].

O NS-2 é um simulador de eventos discretos freeware desenvolvido pelo *Network Research Group* do Laboratório Nacional de Lawrence Berkeley. É de grande utilidade na simulação de redes locais ou redes de grande área de cobertura. Actualmente suporta várias implementações de redes IP e também protocolos de redes como TCP/UDP, FTP, HTTP, etc; tem também disponíveis protocolos de routing, QoS, gestão de filas, modelos de tráfego e suporta ligações físicas ou sem fios. O simulador NS-2 é um produto sempre em desenvolvimento pois cada versão implementa novos protocolos. A principal desvantagem decorrente desta evolução é que a documentação não acompanha tão rapidamente o lançamento das novas versões do NS-2.

O NS-2 é um simulador de domínio livre mas de difícil aprendizagem. Para um utilizador/investigador o estudo da ferramenta toma grande parte do projecto.

Visão geral

O NS-2 é um simulador orientado a objectos escrito em C++ e utiliza um interpretador Otcl como *frontend*. O simulador suporta uma hierarquia de classes em C++ e uma hierarquia de classes dentro do interpretador OTcl que estão relacionadas entre si.

São utilizadas duas linguagens de programação porque o simulador necessita de cada uma delas para uma determinada função. Por um lado, é necessária uma linguagem de programação (C++) eficiente na manipulação de bytes, pacotes, cabeçalhos e grandes quantidades de dados. Por outro lado, as simulações de redes envolvem mudança e variação de parâmetros mas principalmente a rápida exploração de diversos cenários. Neste caso, a velocidade de iteração é importante para uma rápida modificação do modelo e nova simulação.

O NS-2 reúne estes requisitos com duas linguagens, o C++ e o Otcl. Com o C++ a execução é rápida mas a modificação é lenta, tornando-a ideal para programação detalhada de protocolos. O Otcl permite uma rápida modificação de parâmetros de simulação e cenários ideais para simulações.

A figura seguinte mostra as principais partes do simulador em questão:

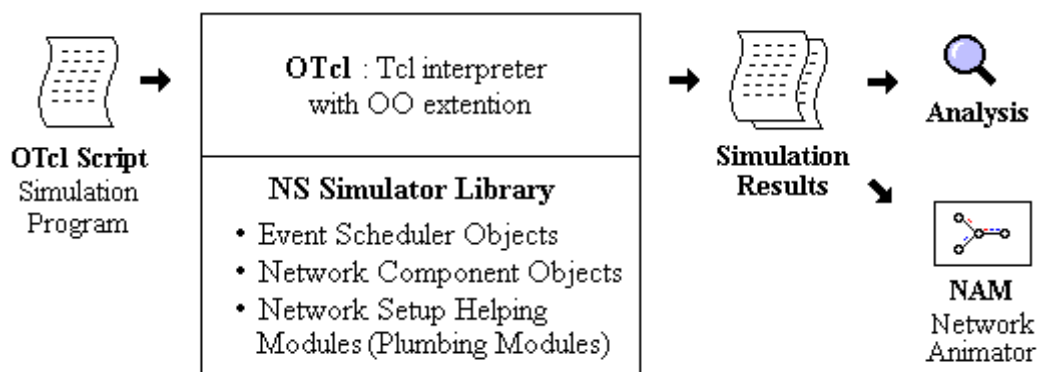


Figura 6 Blocos importantes do Simulador NS-2[7]

Para utilizar este simulador, é necessário programar um script em OTcl.

O NS-2 é assim um interpretador de scripts escritos em Otcl que possui um gestor de eventos. Para além disso é indispensável indicar o início e o término da transmissão de pacotes nas fontes de tráfego através do gestor de eventos.

Um evento no NS-2 possui um ID que é único para cada tipo de pacote e um apontador que controla o evento. Este gere os tempos de simulação e lança os eventos para a fila, invocando os componentes de rede necessários.

Os componentes de rede comunicam entre si passando pacotes. No entanto, isto não consome de tempo de simulação. Todos os componentes de rede que necessitem gastar tempo a tratar um pacote (por exemplo, para simular um atraso), utiliza o gestor de eventos para atribuir o lançamento do pacote na simulação com o atraso necessário.

Sendo o NS-2 um simulador de eventos discretos, o gestor de eventos assume importância.

O NS-2 tem dois tipos de gestores de eventos implementados. Um é um gestor em tempo real e o outro em tempo não real. No gestor em tempo não real estão disponíveis três implementações (lista, pilha e agenda). A implementação em agenda é a que está definida por omissão no simulador. O escalonador em tempo real serve para emulações, permitindo ao simulador interagir com uma rede real (ainda em fase de desenvolvimento).

O funcionamento de um gestor de eventos discretos é descrito na Figura 7.

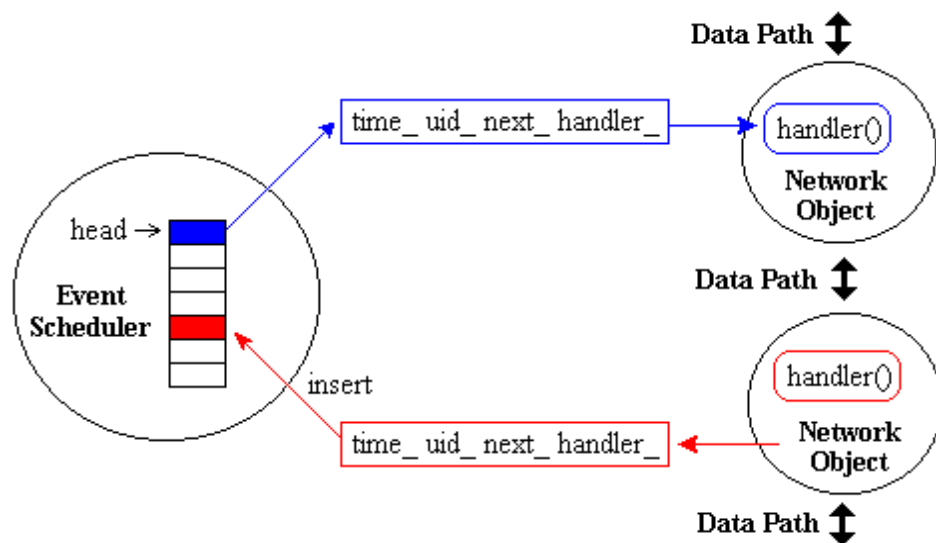


Figura 7 Escalonador de Eventos [39]

Cada evento é inserido no gestor e estes são executados um por um.

Arquitetura do NS-2

O NS-2 é escrito não só em OTcl mas também em C++. Por razões de eficiência, o NS-2 separa a implementação de curso de dados do curso de implementação de controlo. Tendo

em mente a redução do tempo no processamento de pacotes, o gestor de eventos e os componentes básicos de rede são escritos em C++. Este objectos estão disponíveis para o OTcl através do tclcl. É criado assim um objecto Otcl equivalente para cada objecto C++. Deste modo é possível escrever scripts em OTcl e controlar objectos C++.

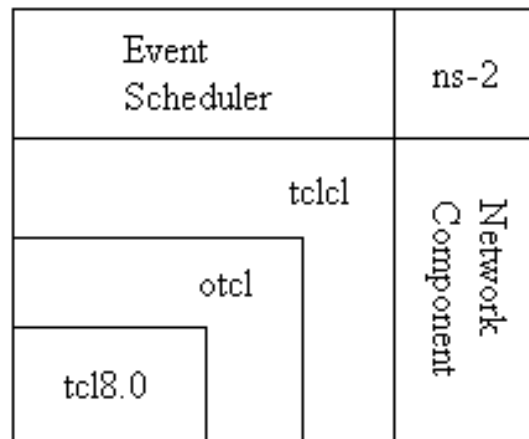


Figura 8 Arquitectura NS-2 [39]

A figura mostra a arquitectura geral do NS. Um utilizador deste simulador deverá começar por escrever e executar simulações em OTcl usando os objectos de simulação da biblioteca OTcl. Os gestores de eventos e a maioria dos componentes de rede são escritos em C++ e disponíveis para o OTcl através de uma ligação implementada usando tclcl. O conjunto destes blocos dá origem ao NS.

Nós e Routing

O nó é uma classe no OTcl e é composto por mais dois objectos Tcl: classificador de endereço e um classificador de porta.

Um nó é composto por um nó de entrada e por classificadores como mostra a figura seguinte. Existem dois tipos de nós no NS-2: *unicast* e *multicast*. Nós *unicast* possuem um classificador que faz routing *unicast* e um classificador de porta. O nó *multicast* é igual ao nó *unicast* com a excepção de este possui um classificador *multicast* que confere routing *multicast*. Por omissão todos os nós são *unicast*.

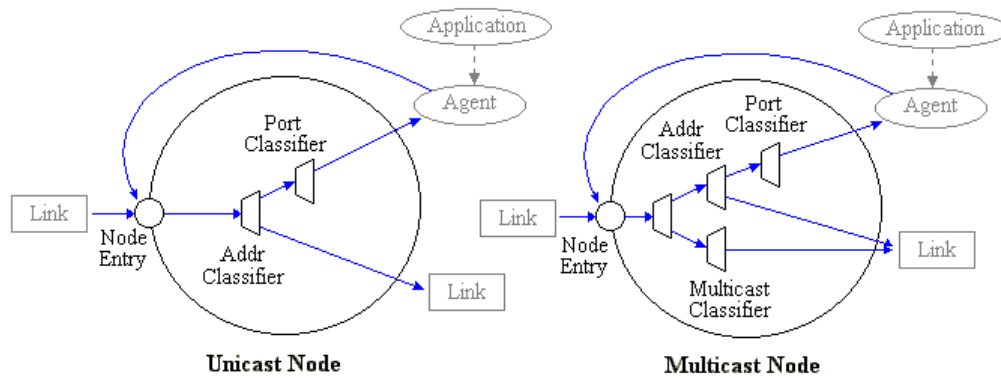


Figura 9 Tipos de nós *unicast* e *multicast* [39]

Ligações

Este é outro aspecto na criação de uma topologia. As ligações entre os nós completam a definição de topologia. Apresenta-se, então, a estrutura mais detalhada de uma ligação na figura seguinte:

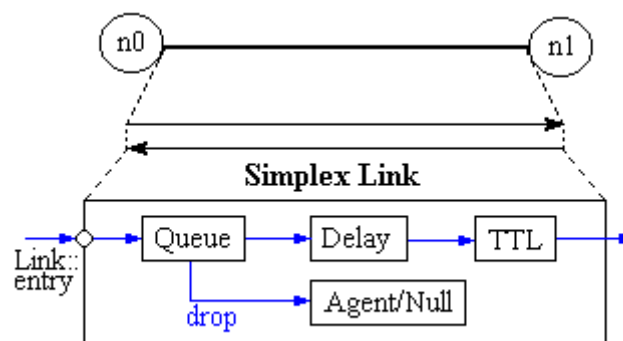


Figura 10 Ligação unidireccional entre nós [7]

Quando um utilizador cria uma ligação duplex – *duplex link* – são criadas 2 ligações *simplex* (unidireccional) em ambas as direcções.

O funcionamento é o seguinte: os pacotes recebidos, depois de libertados pela Queue, passam para o objecto Delay que simula o atraso da ligação. Os pacotes descartados da fila são enviados para o *Null Agent*. Por fim, o bloco TTL calcula o parâmetros *Time to Live* para cada pacote recebido e actualiza esse valor.

Como parte de um trabalho de simulação, na maior parte das vezes queremos efectuar registo da simulação para uma posterior avaliação.

Deste modo, à ligação são acrescentados blocos para a realização de recolha de dados da simulação.

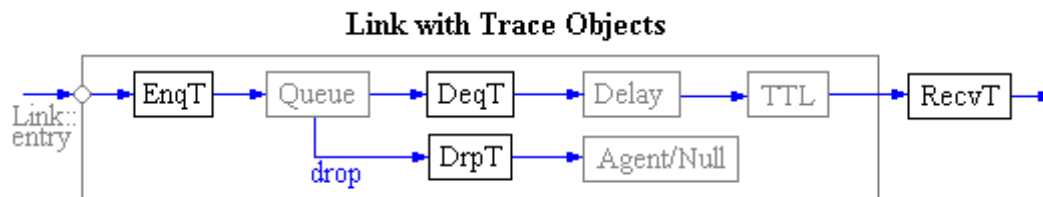


Figura 11 Ligação com recolha de dados [7]

Esses blocos são:

EnqT_ : refere-se ao elemento que analisa pacotes que entram na fila.

DeqT_ : refere-se ao elemento que analisa pacotes que saem da fila.

DrpT_ : refere-se ao elemento que analisa pacotes que se perdem na fila.

RecvT_ : refere-se ao elemento que analisa pacotes recebidos no próximo nó.

Resultados

No código Otcl da simulação é imprescindível inserir as linhas necessárias para que o simulador registre num ficheiro tudo o que é simulado.

Esse registo é um ficheiro chamado *trace file* que, no seu conteúdo, reúne informação da seguinte forma:

event	time	from node	to node	pkt type	pkt size	flags	fid	src addr	dst addr	seq num	pkt id
r : receive	(at to_node)										
+ : enqueue	(at queue)							src_addr : node.port (3.0)			
- : dequeue	(at queue)							dst_addr : node.port (0.0)			
d : drop	(at queue)										


```

r 1.3556 3 2 ack 40 ----- 1 3.0 0.0 15 201
+ 1.3556 2 0 ack 40 ----- 1 3.0 0.0 15 201
- 1.3556 2 0 ack 40 ----- 1 3.0 0.0 15 201
r 1.35576 0 2 tcp 1000 ----- 1 0.0 3.0 29 199
+ 1.35576 2 3 tcp 1000 ----- 1 0.0 3.0 29 199
d 1.35576 2 3 tcp 1000 ----- 1 0.0 3.0 29 199
+ 1.356 1 2 cbr 1000 ----- 2 1.0 3.1 157 207
- 1.356 1 2 cbr 1000 ----- 2 1.0 3.1 157 207
  
```

Figura 12 Exemplo Trace gerado pelo NS-2 [39]

Cada linha deste ficheiro começa pelo evento (+,-,d,r) seguido do tempo em segundos. De seguida, fornece informação do nó de origem e no destino bem como o tipo de pacote e tamanho deste. A informação seguinte (que surge como “-----”) é reservada para indicação de *flags*. Depois temos o flow id, portas de origem e destino, número de sequência e o packet id.

Em suma, toda esta informação aparece de uma forma massiva, podendo cada ficheiro ter mais ou menor dimensão dependendo do tempo de simulação.

Toda essa informação deve ser tratada para que possa ser analisada. Para análise deste ficheiro de trace foram utilizadas scripts em linguagem AWK [36].

O trace file tem, também neste caso, informação relativa ao que se passa no meio sem fios.

O tipo de informação que surge no novo trace difere um pouco e tem a forma apresentada na figura seguinte:

```
r -t 5.275742000 -Hs 3 -Hd 4194306 -Ni 3 -Nx 0.00
-Ny 0.00 -Nz 0.00 -Ne -1.000000 -Nl MAC -Nw --- -
Ma 0 -Md 0 -Ms 8 -Mt 0 -Is 0.2 -Id 4194306.0 -It
udp -Il 1014 -If 3 -Ii 1734 -Iv 30

d -t 5.275742000 -Hs 4 -Hd 4194306 -Ni 4 -Nx 0.00
-Ny 0.00 -Nz 0.00 -Ne -1.000000 -Nl MAC -Nw --- -
Ma 0 -Md 0 -Ms 8 -Mt 0 -Is 0.2 -Id 4194306.0 -It
udp -Il 1038 -If 3 -Ii 1734 -Iv 30

d -t 5.275742000 -Hs 5 -Hd 4194306 -Ni 5 -Nx 0.00
-Ny 0.00 -Nz 0.00 -Ne -1.000000 -Nl MAC -Nw --- -
Ma 0 -Md 0 -Ms 8 -Mt 0 -Is 0.2 -Id 4194306.0 -It
udp -Il 1038 -If 3 -Ii 1734 -Iv 30

d -t 5.275742000 -Hs 6 -Hd 4194306 -Ni 6 -Nx 0.00
-Ny 0.00 -Nz 0.00 -Ne -1.000000 -Nl MAC -Nw --- -
Ma 0 -Md 0 -Ms 8 -Mt 0 -Is 0.2 -Id 4194306.0 -It
udp -Il 1038 -If 3 -Ii 1734 -Iv 30
```

Constata-se que a quantidade de informação é maior. À semelhança do trace para acesso físico começa pelo tipo de evento (s, r, d) seguido do tempo de simulação t. Cada campo é precedido por uma *tag* que fornece a indicação do tipo de campo.

Os campos -Hs e -Hd representam o id do nó em questão e o próximo nó. “-Ni” é o id do nó, “-Nx/y/z” as coordenadas (aplicável só no caso de mobilidade), “-Ne/l/w” a energia, nível do trace e tipo de evento.

Informação do pacote no nível MAC:

Campos iniciados com a tag "-M"

-Ma: duração

-Md: endereço ethernet destino

-Ms: endereço ethernet inicial

-Mt: tipo ethernet

Informação do pacote a nível IP:

Tags dos campos com "-I".

-Is: endereço inicial seguido do número da porta

-Id: endereço destino.porta destino

-It: tipo de pacote

-Il: tamanho do pacote

-If: id do flow

-Ii: id único

-Iv: valor TTL

Futuramente, o trace do NS-2 será este quer se trate de acesso físico ou sem fios.

Geração de números Aleatórios

O simulador inclui um conjunto de funções matemáticas para a geração de números aleatórios e sua integração.

O componente RNG é utilizado numa variedade de outros componentes no NS-2, como geração de números aleatórios para diferentes tipos de distribuições matemáticas (uniforme, exponencial, pareto, normal, lognormal, etc). Deste modo, a manipulação do

componente RNG toma elevada relevância no simulador, visto estar presente nas principais funções do NS-2.

A classe RNG contém uma implementação de um gerador múltiplo recursivo combinado, chamado MRG32k3a, proposto por Lécuyer [17].

O gerador oferece 1.8×10^{19} *streams* independentes de números aleatórios e cada um com 2.3×10^{15} *substreams*. Cada *substream* tem uma extensão de 7.6×10^{22} números. O gerador tem na totalidade 3.1×10^{57} números aleatórios. A Figura 13 mostra como o gerador é composto.

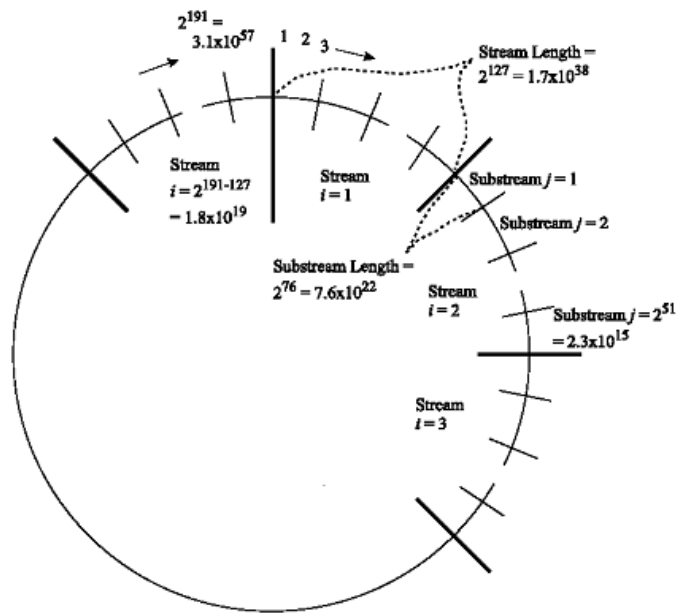


Figura 13 *Streams* e *substreams* do gerador de números aleatórios [7]

O objecto RNG é criado por omissão no simulador. Se foram utilizadas múltiplas variáveis numa simulação deve ser criado um objecto RNG separado. Quando um novo objecto RNG é criado, este é automaticamente associado ao início do próximo *stream* de números aleatórios. Esta forma de implementação permite um máximo de 1.8×10^{19} variáveis aleatórias.

Em situações em que é necessário fazer várias simulações com os mesmos parâmetros, mas com sementes diferentes, em cada replicação deve ser escolhido um *substream* diferente para assegurar que os números aleatórios gerados são diferentes. Este modo permite ter um máximo de 2.3×10^{15} replicações independentes.

3.2. SIMULAÇÃO WIMAX – MODELOS ADOPTADOS

3.2.1. MÓDULO WIMAX

Existem actualmente três modelos gratuitos para simular a tecnologia de acesso WIMAX com o simulador NS-2.

A escolha do módulo WIMAX foi baseada num estudo [8] do qual se elaborou uma síntese dos módulos existentes para o simulador NS-2. O resumo desse trabalho é apresentado na Tabela 1:

	NDSL	UNICAMP	NIST
Facilidade de instalação para a versão 2.32 do ns	Complexo	Muito Complexo	Simple
Complexidade do código C++	Muito Complexo	Complexo	Simple
Complexidade do código Tcl	Simple	Complexo	Simple
Bugs encontrados	Muitos	Poucos	Poucos
Mecanismos de suporte de QoS	Existente	Existente	Sem escalonador
Mobilidade	Inexistente	Inexistente	Existente
Manual de utilização	Inexistente	Inexistente	Existente
Facilidade de criação de cenários	Complexo	Complexo	Complexo
Análise dos resultados	Complexo	Complexo	Simple

Tabela 1 Estudo comparativo de módulos WIMAX para NS-2 [8]

Para este trabalho a mobilidade não era um requisito, mas sim o mecanismo de suporte QoS.

Comparativamente ao NDSL [11] e ao UNICAMP [13], a escolha recaiu sobre o módulo mais recente a ser disponibilizado para o NS-2 – UNICAMP, visto ter menos *bugs* e possuir mecanismos QoS. Preenche os requisitos para este trabalho tendo, contudo, algumas desvantagens como maior complexidade na instalação/integração no NS-2 e a complexidade do código C++.

Descrição do módulo WIMAX

A implementação deste módulo centrou-se na camada MAC e seus mecanismos de alocação de largura de banda e suporte QoS. Este módulo suporta o modo TDD – *Time*

Division Duplex e topologia PMP (Ponto-Multiponto). O modelo de acesso sem fios WIMAX foi desenvolvido por [12] com base noutra projecto, no qual se modelou a norma DOCSIS, anteriormente criado para o simulador NS-2. Contudo, foram feitas as modificações necessárias para que fossem cumpridos os requisitos da norma 802.16.

Entre essas modificações destacam-se:

- implementação do serviço nrtPS para o tráfego no *uplink*;
- implementação dos cinco tipos de serviço para o tráfego no *downlink*;
- alteração da interface entre a camada MAC e a camada física para que o módulo WIMAX utilizasse o meio sem fios disponível no NS-2;
- implementação da divisão do tempo em frames e dos frames em subframe *downlink* e subframe *uplink*;
- inclusão dos parâmetros de QoS latência máxima, para o serviço rtPS, e largura de banda mínima, para os serviços rtPS e nrtPS.

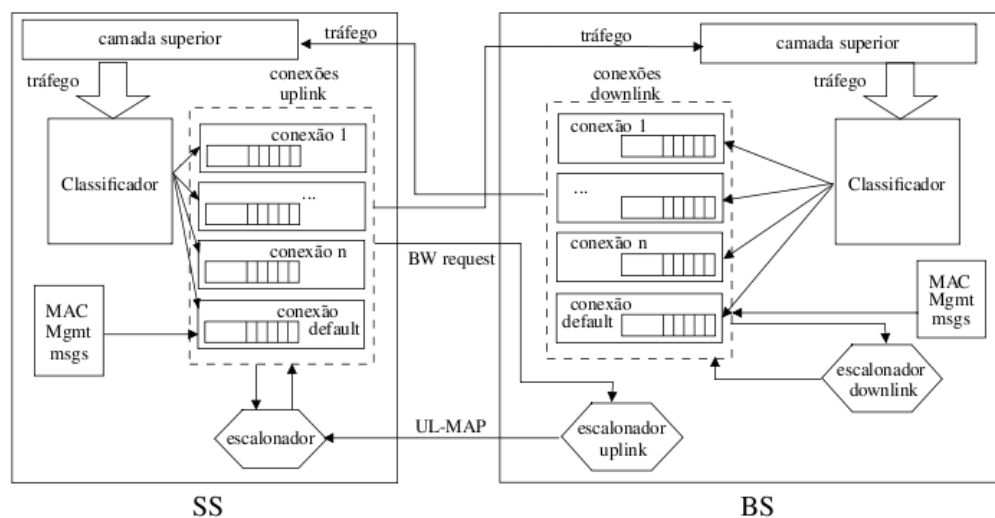


Figura 14 Modelo esquemático do módulo WIMAX [12]

A Figura 14 apresenta de uma forma esquemática o módulo WIMAX. O tráfego proveniente da camada superior, na SS ou na BS, passa pelo classificador e, baseado no tipo de tráfego e do par origem e destino, é colocado no fluxo de serviço apropriado ficando, assim, associado a uma ligação. Os fluxos de serviço são definidos pelo utilizador determinando assim os requisitos de QoS.

3.2.2. MÓDULO VOIP

O módulo de VOIP utilizado neste trabalho foi desenvolvido na Universidade de Pisa, pelo grupo de Redes e Computadores do Departamento de Engenharia e Informação [14] e [15].

Este módulo foi desenvolvido devido à necessidade da existência de um modelo de VOIP mais completo.

A integração deste módulo no NS-2 correu praticamente sem problemas visto que o *patch* disponibilizado pelos autores era para a versão de NS-2 utilizada neste trabalho.

Implementação do módulo no simulador NS-2

As funcionalidades implementadas são apresentadas separadamente para o emissor e o receptor.

Para o emissor estão implementadas as seguintes funções: *codec* customizável que gera frames de voz e um multiplexer, que agrega várias frames de voz em um único *payload*.

Do lado de recepção: algoritmo configurável para o *playout buffer* e um desmultiplexer.

Do ponto de vista funcional a Figura 15 resume todos os blocos que constituem este módulo VOIP.

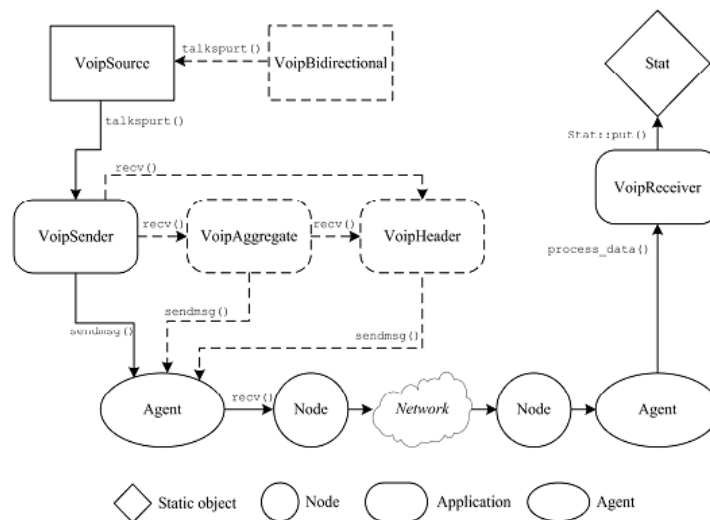


Figura 15 Blocos funcionais do módulo VOIP [15]

O bloco *VoipSource* gera todas as frames de voz. Estas são automaticamente configuradas para cada tipo de cenário como: conversação de um-para-um, múltiplos intervenientes

numa conferência, leitor/audiência. No início de cada conversação são enviados os períodos de conversa/silêncio. Depois o *VoipSender* inicia a geração de itens *VoipFrame* cujo tamanho e velocidade de geração dependem do *codec* utilizado. Os *codecs* mais frequentes, como o G.711e GSM.AMR, são suportados pelo *VoipSender*. Este módulo tem uma estrutura que permite adicionar mais *codecs* no futuro.

Caso os blocos de multiplexagem e/ou *header* estejam activos, as frames resultantes são diferentes. Se a multiplexagem estiver activa, o objecto *VoipAgregate* é associado ao bloco *VoipSender*, que faz uma chamada à função *recv()* sempre que haja uma frame para passar para o bloco de multiplexagem. O objecto *VoipAgregate* espera um determinado número de frames e depois empacota-os num objeto *VoipPayLoad* e envia-os para o *VoipHeader* (se estiver activo) ou directamente para o Agente.

O *VoipHeader* recebe um objecto *VoipPayLoad* do *VoipSender* ou *VoipAgregate*, dependendo se a multiplexagem estiver activa. O *VoipHeader* adiciona cabeçalho RTP/UDP/IP ao *VoipPayLoad* e suporta assim a compressão do cabeçalho. Depois o *payload* é enviado para o agente.

O agente encapsula o objecto *VoipPayLoad* num pacote NS-2 e é este pacote que atravessa a rede de simulação susceptível de atrasos, reordenação ou perdas.

Os pacotes que eventualmente não sejam perdidos são recebidos pela aplicação *VoipReceiver*. Este recebe todos os objectos *VoipFrame* listados no *VoipPayLoad* e inicia o *buffering* do *playout*. As frames VOIP são simulados como pacotes sem conteúdo propriamente dito (como é, aliás, usual em simulações com NS-2), ou seja, cada pacote NS-2 que viaja pela rede simulada não contém a informação que um *encoder* VOIP produziria num ambiente real.

3.2.3. MÓDULO VoD – VIDEO ON DEMAND

O modelo de tráfego *Video on Demand* utilizado neste trabalho foi desenvolvido na Universidade do Estado do Arizona [19] . O *stream* de vídeo usado como *input* para o simulador resulta de uma captura segundo o modelo descrito na Figura 16, produzindo um ficheiro de dados:

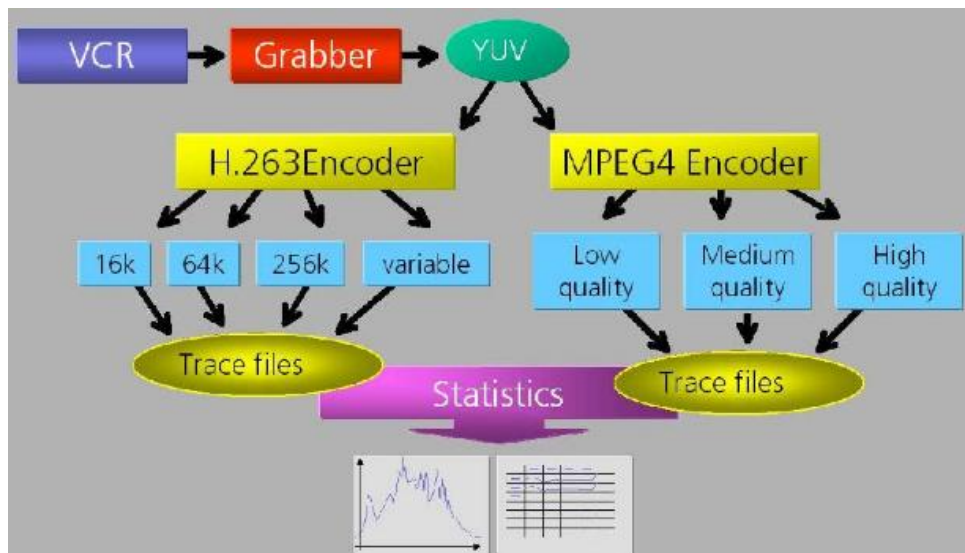


Figura 16 Modelo para produzir ficheiros *Trace* de Vídeo [18]

Para cada vídeo foi capturada a informação YUV (modelo para representar a cor) utilizando o software *bttvgrab* (Versão 0.15.10) [20].

A informação obtida pelo software em YUV serve de *input* ao *encoder* MPEG-4. Para isso foi utilizado o MOMUSYS MPEG-4 [32] para codificação. Esta foi feita em três níveis de qualidade: baixo, médio, e alto produzindo ficheiros distintos. O ficheiro produzido por esta ferramenta serve de *input* para o simulador NS-2. Várias codificações estão disponíveis no *site* [18] para *download* livre, em várias qualidades e com fontes de vídeo distintas. Essas fontes são diferentes segmentos de vídeo por exemplo: filmes, documentários, noticiários etc. Este tipo de modelo de tráfego baseado ficheiros *trace* é o mais aproximado ao real porque modeliza na perfeição as propriedades do ficheiro de vídeo original, como *codec*, conteúdo e tipo de *encoder*. Seguem-se alguns exemplos de ficheiros *trace* utilizados neste trabalho assim como as respectivas características:

Vídeo	Taxa Média de Transferência (Mbps)	Taxa Máxima de Transferência (Mbps)	Variância do tamanho dos frames	Hurst
Friends vol4 DVD	0.181544	1.175520	71620243.645425	0.777175
Lecture Gupta	0.181544	2.218080	71620243.645425	0.777175
Baseball	0.144776	1.859760	38430999.822669	0.684708
Lady Tramp DVD	0.048951	0.910080	8282030.584567	0.813108
Ice Age DVD	0.065940	1.058400	11347671.977999	0.834963
Dreamcatcher DVD	0.063146	0.932400	8685118.764458	0.700975
Aladdin	0.209843	2.242800	72480467.128562	0.876541

Figura 17 Características de cada sequência de vídeo [13]

Os *traces* foram codificados em resolução QCIF (174*144 pixéis), que são os formatos de resolução de imagem mais utilizados para simulações.

3.2.4. MÓDULO HTTP

O modelo utilizado neste trabalho foi desenvolvido por Kleber Vieira Cardoso no âmbito de uma Pós-Graduação na Universidade Federal do Rio de Janeiro [22]. De entre os modelos HTTP disponíveis para o simulador NS-2, este diferencia-se pelo método de implementação (método agregado).

A maior parte dos modelos de tráfego HTTP desenvolvidos concentraram-se em descrever o comportamento de cliente Web [23][24][25]. Todos eles diferenciam-se pelos parâmetros escolhidos como tempo de processamento de uma página. Normalmente os tempos de espera, processamento de página e tamanho de transferência da página visitada são descritos por distribuições matemáticas. Por outro lado, quando se tenta caracterizar um conjunto de clientes Web (por exemplo várias visitas distintas de páginas Web), torna esta aproximação muito complexa com definição de um número elevado de parâmetros.

O modelo proposto pelo autor conta com número reduzido de parâmetros capazes de reproduzir uma determinada carga na rede. O modelo é chamado de agregado porque tem como objectivo representar a carga que seria gerada por um determinado número de clientes Web (não especificado).

Parâmetros modelizados

Numa abordagem analítica para a construção do modelo, temos que ter em consideração que existem parâmetros a definir como:

- Tamanho de página ou objecto: este é o parâmetro mais avaliado aquando da construção de um modelo, visto que o tamanho do objecto tem uma relação directa com o funcionamento do controlo de congestionamento do TCP. A distribuição Pareto é a mais utilizada mas existem trabalhos em que utilizam Lognormal ou híbrida (Pareto + Lognormal).
- Números de objectos por página: define a quantidade de elementos que compoem uma página Web. Não existe muito consenso nos valores assumidos para este parâmetro. Algumas distribuições podem descreve-lo: Gama, Lognormal e Pareto.

- Tempo de visualização: descreve o comportamento típico de um utilizador Web enquanto acede a um conteúdo Web, ou seja, o tempo consumido ao ler o texto de uma página, preencher um formulário, visualizar um vídeo, etc. É um parâmetro difícil de caracterizar devido ao factor humano e devido à diversidade de conteúdos a que um utilizador acede. Distribuições escolhidas: Weibull, Gama e Lognormal.

- Tempo de silêncio: tempo entre sessões ou páginas, isto é, o período entre a recepção do último objecto até ao próximo pedido do utilizador. Este é também um parâmetro difícil de avaliar. Nesta situação podem ocorrer tempos nulos, pois o utilizador pode gerar um novo pedido antes do actual pedido ser atendido por completo. Para descrever este parâmetro usam-se distribuições: Pareto ou o processo de chegada Poisson.

- Tamanho dos pedidos: apesar da maioria dos trabalhos descartarem este parâmetro, quem o implementa aumenta a qualidade do seu modelo. Exemplos de distribuições usadas: Lognormal e Bimodal.

Em estudos sobre HTTP, o tamanho da página/objecto é o aspecto mais abordado [22]. Isto ocorre, sobretudo, pela influência desta característica em todos os modelos. O autor modeliza este parâmetro \bar{A} e, no módulo desenvolvido, podem ser seleccionadas as distribuições a usar na simulação. A tabela seguinte mostra as diferentes implementações de cada modelo para caracterizar o parâmetro \bar{A} .

Distribuição	Configuração	Nome	Referência
Pareto	média - 4100 <i>shape</i> - 1.95	HTTP-1	[29]
Pareto	média - 4100 <i>shape</i> - 1.35	HTTP-2	[29]
Lognormal	média - 4827 desvio padrão - 41008	HTTP-3	[27]
híbrida: Pareto - 7%, Lognormal - 93%	média - 1463000 <i>shape</i> - 1.1 média - 27600 desvio padrão - 59714	HTTP-4	[28]
híbrida: Pareto - 12%, Lognormal - 88%	média - 10558 <i>shape</i> - 1.383 média - 7247 desvio padrão - 28765	HTTP-5	[30]

Tabela 2 Distribuições do parâmetro \bar{A}

4. SIMWIMAX - APLICAÇÃO DE SIMULAÇÃO WIMAX

Neste capítulo é apresentada a aplicação SimWimax para auxiliar no dimensionamento de redes e cenários de partilha de acessos WIMAX. Inicialmente, o capítulo descreve a arquitectura geral da aplicação. De seguida, é referida a *frontend web*, suas funcionalidades, opções e páginas que a constituem. Posteriormente, é explicada a forma como a aplicação SimWimax interage com o simulador NS-2.

A aplicação SimWimax é no geral um *frontend* gráfico *web* para o simulador NS-2 que, para além de executar o simulador, faz tratamento de dados em *background* e apresenta-os em forma gráfica. A aplicação permite simular a tecnologia WIMAX, configurar os seus parâmetros bem como dos tipos de tráfego.

4.1. APRESENTAÇÃO GENÉRICA DA APLICAÇÃO

De uma forma sumária, a figura seguinte mostra os principais blocos da aplicação SimWimax:

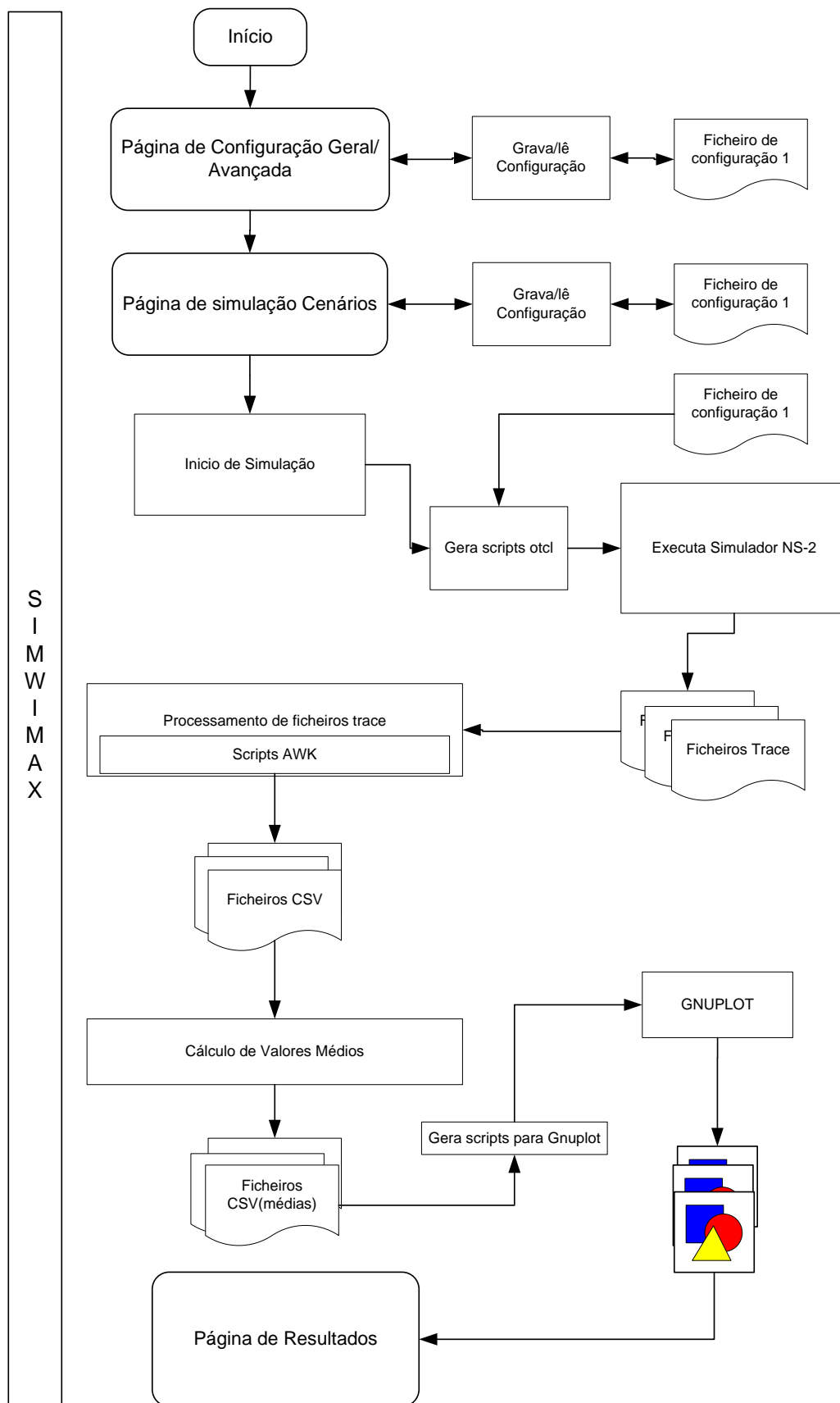


Figura 18 Blocos da aplicação SimWimax

A aplicação SimWimax disponibiliza um *frontend* gráfico no qual o utilizador parametriza a sua simulação. Antes de a iniciar o utilizador tem um leque de opções a configurar.

Nas páginas de configuração geral/avançada o utilizador pode definir variados aspectos relacionados com os modelos de tráfego e com o simulador NS-2. Todas as informações de configuração são gravadas num ficheiro txt. Quando se dá início a uma simulação a aplicação gera o script tcl, baseado no ficheiro de configuração) para o NS-2. O simulador é executado dando origem a ficheiros *trace* (fruto das diversas simulações).

Na fase seguinte, dá-se início ao tratamento de dados sendo executados *scripts* em AWK para o efeito. O output dos scripts AWK são ficheiros CSV com informação de taxa de transferência e atraso. No bloco seguinte são calculados valores médios para a taxa de transferência e atraso. Depois são criados scripts para a geração de gráficos no Gnuplot.

Após execução do Gnuplot são obtidos gráfico em formato png sendo estes apresentados na Página de Resultados.

4.2. CENÁRIOS DE APLICAÇÃO RESIDENCIAL

- Cenário de aglomerado ou bairros de habitações

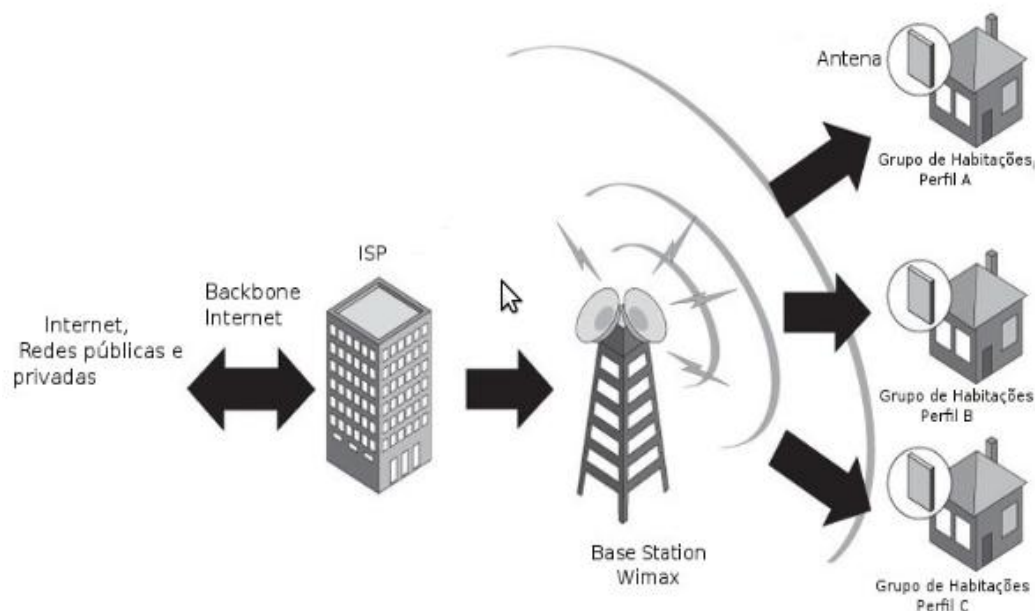


Figura 19 Arquitectura do Cenário de aglomerado de habitações

A figura anterior mostra a arquitectura do cenário residencial em que a topologia é ponto-multiponto. Existe um ISP ligado fisicamente à BS e as estações cliente SS estão

distribuídas geograficamente em torno da BS. Cada habitação possui uma antena de acesso.

- Cenário de edifício comum

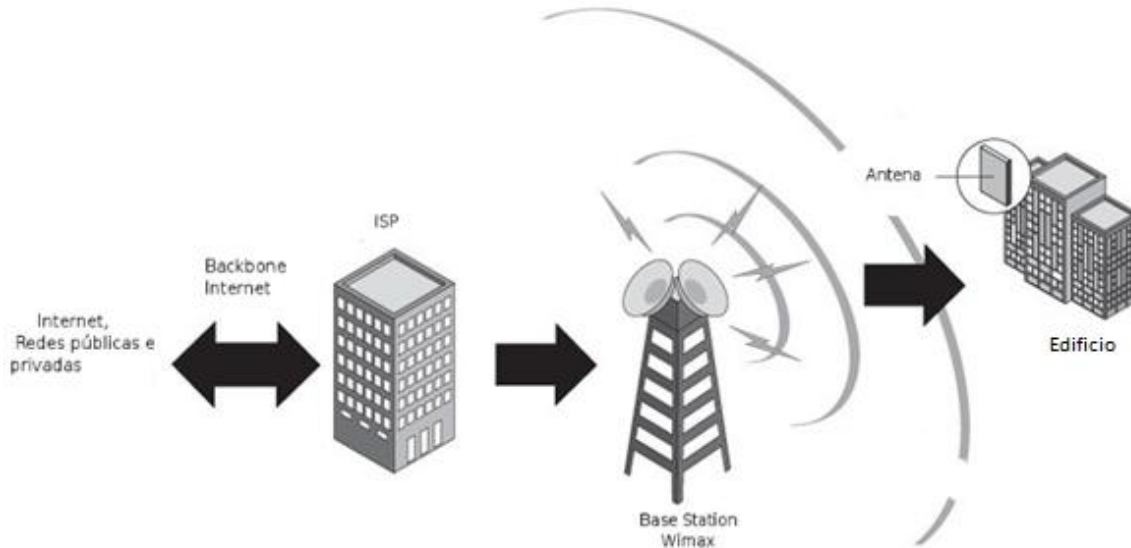


Figura 20 Arquitectura do Cenário de edifício comum

Na Figura 20, a topologia presente é ponto-a-ponto. Entre o ISP e a BS existe uma ligação física de alto débito. A estação cliente agrega um número de clientes superior ao Cenário Residencial. Cada SS possui uma antena.

4.3. PERFIS DE UTILIZADOR

Nas páginas de simulação de cenários residenciais (aglomerado de habitações ou edifício comum) pode ser feito o agrupamento dos utilizadores em perfis A, B ou C.

Para o caso aglomerado de habitações, cada habitação é uma SS. Dentro do mesmo perfil, todas as SS terão as mesmas características de tráfego definidas nas percentagens. O número de SS em cada perfil é dado pelo parâmetro “Habitações”. A utilização de perfis de utilizadores permite, na mesma simulação, definir diferentes tipos de utilizadores. Por exemplo, utilizadores de diferentes faixas etárias ou de diferentes localizações geográficas utilizam os recursos de forma diversa. Então, na aplicação SimWimax estão disponíveis três perfis de utilizadores bem como factores de utilização (percentagens) dos fluxos de tráfego para cada um deles.

Os parâmetros *Upload* e *Download* representam a largura de banda aplicada a cada habitação.

Serviço VOIP

A aplicação VOIP permite a simulação de uma conversa telefónica. Para melhor modelizar a conversa telefónica, foram implementados algoritmos para representar os períodos de silêncio e período de conversação. Os períodos de tempo entre chamadas e a duração das chamadas são modelizados por distribuições:

```
# duracao_ distribuição exponencial aleatoria

set duracao_($j) [new RandomVariable/Exponential]
$duracao_($j) set avg_ $percvoip
$duracao_($j) use-rng $duracaoRNG($j)

# intervalo_ distribuição uniforme aleatoria

set intervalo_($j) [new RandomVariable/Uniform]
$intervalo_($j) set min_ 1
$intervalo_($j) set max_ $percvoip
$intervalo_($j) use-rng $intervaloRNG($j)
```

O período de tempo entre duas chamadas é descrito por uma distribuição uniforme aleatória com valores mínimos e máximos. A duração das chamadas é modelizada por uma variável com distribuição exponencial.

A geração destes tempos é feita em função do factor de utilização definido pelo utilizador na página de simulação do cenário aglomerado/bairro de habitações ou edifício comum.

Para um dado perfil de utilizadores, o período de tempo entre duas chamadas e a duração das chamadas são diferentes. Por exemplo, dois clientes, da mesma SS ou outra, com a percentagem de tráfego definida, terão valores para intervalo entre chamadas e tempo de conversação diferentes para cada utilizador, dado o factor aleatório. São atribuídos diferentes *substreams* de valores aleatórios.

Serviço VoD – Video on Demand

Neste modelo o utilizador desta ferramenta pode escolher na página de configuração geral o tipo de sequência de vídeo (seja filme de animação/acção ou anúncios publicitários) a utilizar na simulação. As próprias características da sequência de vídeo irão afectar a rede de algum modo. Este modelo utiliza como protocolo de transporte o Udp, sendo este usual

para este tipo de aplicações. O factor de utilização é descrito de forma análoga ao serviço VOIP.

Serviço HTTP

No modelo HTTP poderão ser configuradas as distribuições a aplicar para a geração deste tráfego. Como foi visto anteriormente, o parâmetro \bar{A} caracteriza o tamanho de médio de uma ligação web sendo este modelizado por várias distribuições. A escolha deste parâmetro é feita na página de configuração geral da aplicação SimWimax.

4.4. FRONTEND GRÁFICO

O *frontend* gráfico foi desenvolvido utilizando como linguagem de programação o PHP [33]. A escolha do PHP consistiu em vários factores como: software de uso livre, linguagem de programação simples, permitindo rápido desenvolvimento e aprendizagem fácil. Como servidor *web* foi utilizado o Apache. Foi instalado um *bundle* de software chamado LAMP [34] que contém já os dois softwares necessários previamente configurados. Na figura seguinte pode-se visualizar como as diversas páginas estão organizadas na aplicação SimWimax.

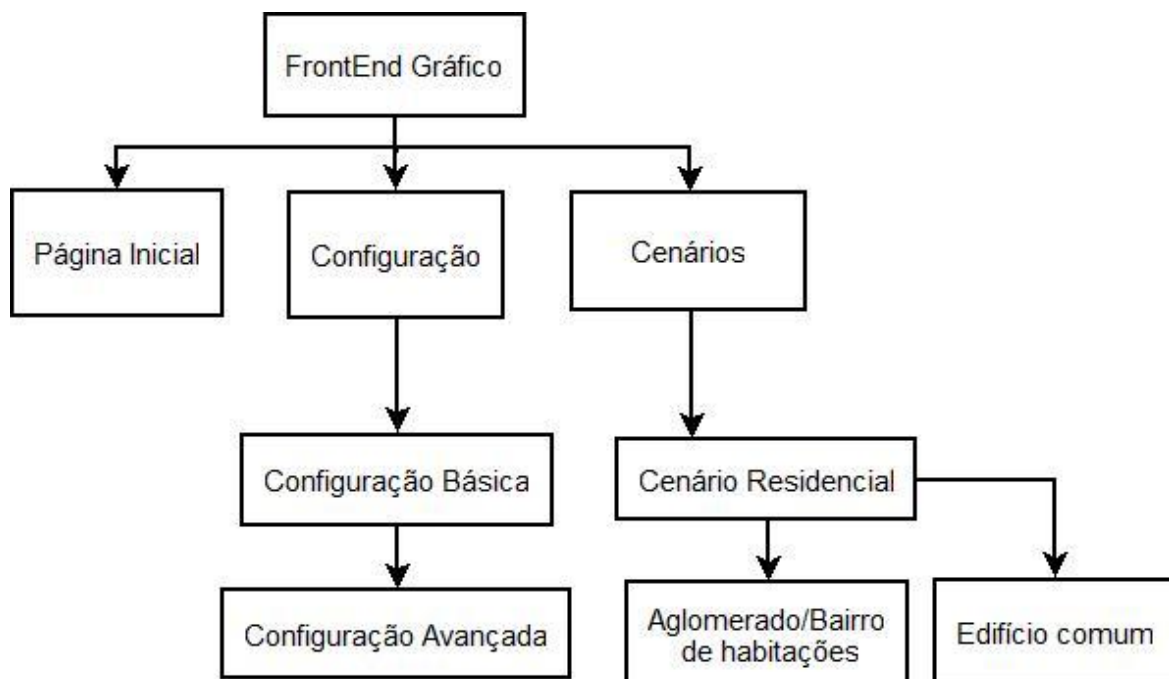


Figura 21 Páginas *web* da aplicação SimWimax

As páginas Inicial, Configuração e Cenários são acedidas através de um menu horizontal situado no topo de página como mostra a figura seguinte.

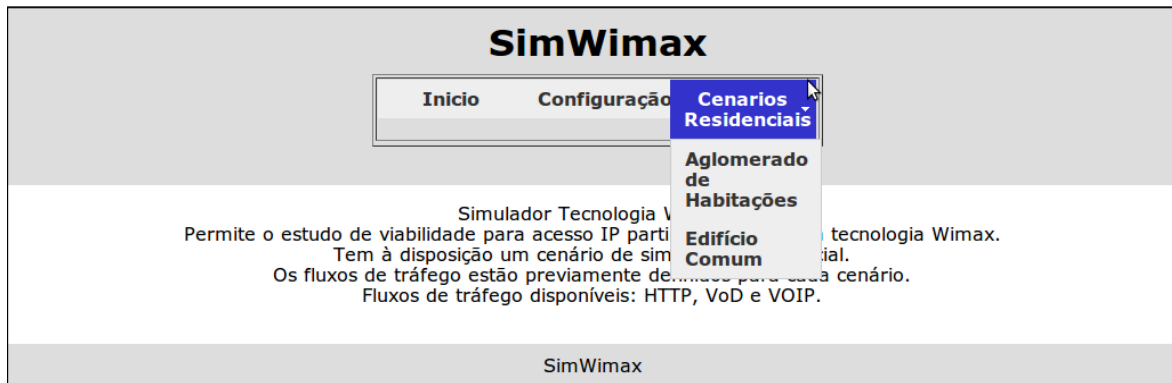


Figura 22 Menu principal SimWimax

O acesso à Configuração Básica é feito directamente pelo menu topo e através desta é possível entrar na página de configuração Avançada.

Na Figura 23 e Figura 24 mostram-se as páginas de Configuração Básica e Avançada.

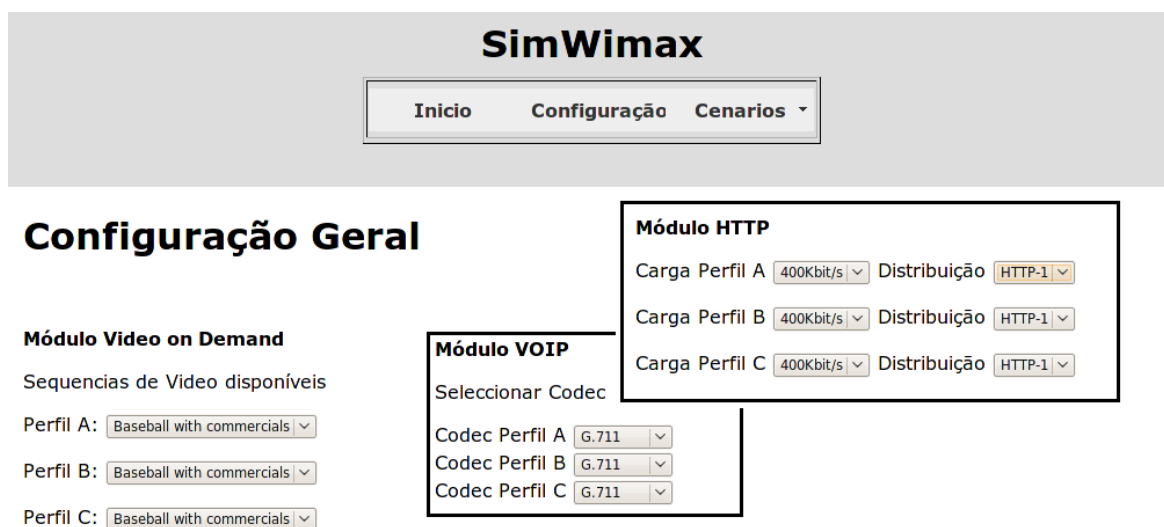


Figura 23 Página de Configuração Básica SimWimax



Figura 24 Página de Configuração Avançada SimWimax

Na página de Configuração Básica é possível configurar as sequências de vídeo a utilizar para cada perfil de utilizadores na simulação, Figura 25.

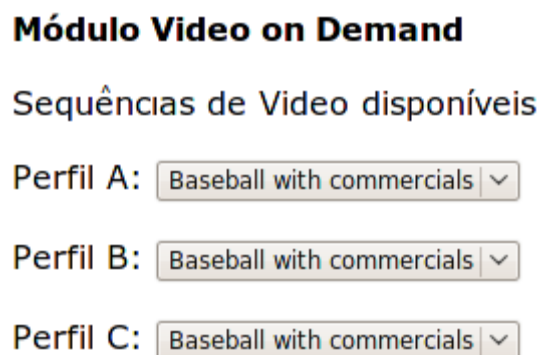


Figura 25 Opção de configuração módulo VoD

Também se pode configurar o *codec* usado para cada perfil de utilizadores na Figura 26.

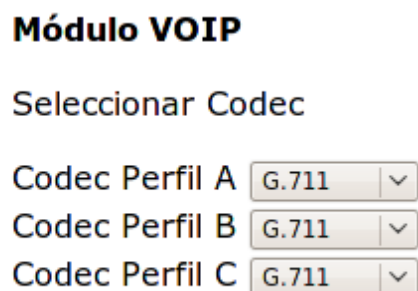


Figura 26 Opções de configuração módulo VOIP

O módulo HTTP pode ser parametrizado como mostra a Figura 27:

Módulo HTTP

Carga Perfil A	<input type="text" value="400Kbit/s"/>	Distribuição	<input type="text" value="HTTP-1"/>
Carga Perfil B	<input type="text" value="400Kbit/s"/>	Distribuição	<input type="text" value="HTTP-1"/>
Carga Perfil C	<input type="text" value="400Kbit/s"/>	Distribuição	<input type="text" value="HTTP-1"/>

Figura 27 Opções de configuração módulo HTTP

O tempo de simulação e semente base para a simulação também podem ser definidos como é ilustrado na Figura 28.

Parâmetros ns-2

Semente:

Tempo de simulação (s):

Configuração Avançada

Figura 28 Opções configuração geral SimWimax

Na página de configuração avançada é permitido ao utilizador do SimWimax parametrizar aspectos relacionados com o QoS do sistema, Figura 29.

Configuração avançada

Parametros QoS	
UGS	
UGS Grant Interval:	<input type="text" value="0.020"/>
UGS Grant Size:	<input type="text" value="66"/>
RTPS	
RTPS Grant Interval :	<input type="text" value="0.020"/>
RTPS Delay (segundos):	<input type="text" value="0.100"/>
RTPS Aggregate Request Interval:	<input type="text" value="5"/>
RTPS Min BW:	<input type="text" value="200000"/>

ERTPS

ERTPS Grant Interval :

ERTPS Grant Size:

NRTPS

NRTPS Grant Interval :

NRTPS Min BW:

NRTPS Aggregate Request Interval:

BE

BE Grant Interval :

BE Aggregate Request Interval:

Figura 29 Opções de configuração avançada no SimWimax

A aplicação SimWimax tem valores já pré-definidos para cada parâmetro. Cabe ao utilizador configurá-los de acordo com a simulação pretendida. Todos os parâmetros alterados nas páginas de configuração deverão ser gravados através do botão “Gravar” no final de cada página. Assim os valores são guardados para que na próxima simulação façam efeito.

Caso o utilizador altere os parâmetros e queira voltar aos valores predefinidos, pode fazê-lo através do botão “Reset” situado junto ao botão “Gravar” (Figura 28). A Figura 30 mostra a página de simulação – Cenário Residencial – Aglomerado de habitações

Cenário Residencial

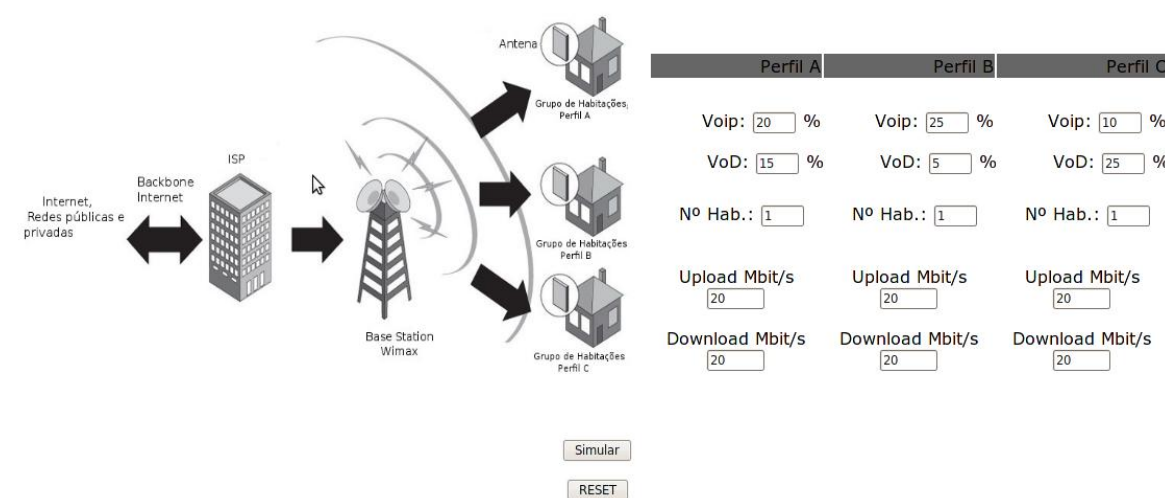


Figura 30 Página de simulação Cenário Residencial - Aglomerado de habitações

Aqui são colocados à disposição do utilizador mais alguns parâmetros de configuração dos módulos. Na Figura 31 constam os parâmetros que o utilizador da aplicação pode alterar.

Perfil A	Perfil B	Perfil C
Voip: 20 %	Voip: 25 %	Voip: 10 %
VoD: 15 %	VoD: 5 %	VoD: 25 %
Nº Hab.: 1	Nº Hab.: 1	Nº Hab.: 1
Upload Mbit/s 20	Upload Mbit/s 20	Upload Mbit/s 20
Download Mbit/s 20	Download Mbit/s 20	Download Mbit/s 20

Figura 31 Parâmetros de configuração Cenário Aglomerado de habitações

Os valores da figura para cada tipo de tráfego e perfil são factores de utilização. Por exemplo, se for definido 20% para o tráfego VOIP significa que, 20% do tráfego gerado por esse perfil será VOIP. O mesmo raciocínio aplica-se para os restantes valores de percentagens.

Os parâmetros *Upload* e *Download* representam a largura de banda aplicada a cada habitação. Os valores de largura de banda são definidos em Mbit/s.

Por fim o número de habitações, em que cada habitação é caracterizada por diferentes tipos de tráfego.

Para iniciar a simulação basta clicar no botão “Simular”. Após terem sido finalizadas todas as rotinas, o utilizador é direccionado para a Página de Resultados. Nesta página são apresentados os resultados sob a forma de gráficos. A informação dada ao utilizador é por perfil e por cada tipo de tráfego. Para todos os tipos de tráfego é apresentado um gráfico da taxa de transferência e um para o atraso.

A Figura 32 é um exemplo de uma Página de Resultados.

Resultados Residencial

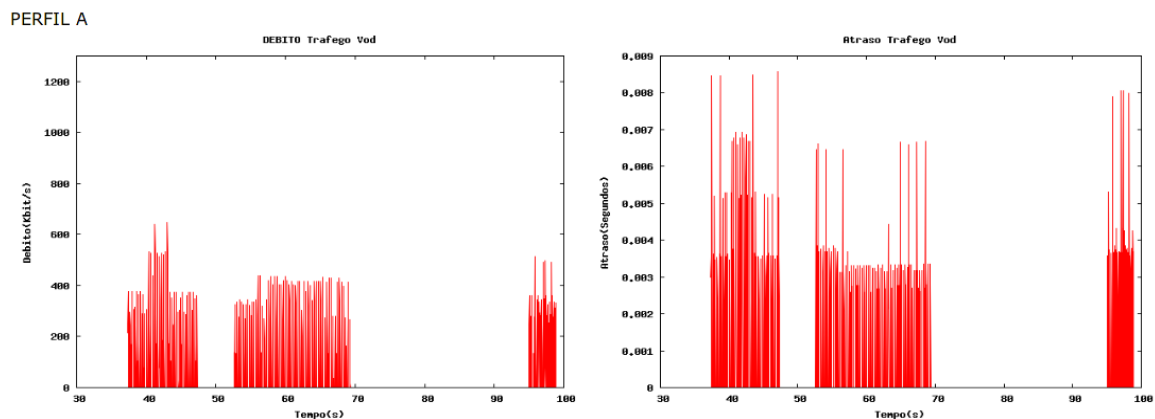


Figura 32 Exemplo da página de resultados do SimWimax

Esta aplicação tem em *background* um processamento pesado de dados. Este peso deriva directamente do processo de simulação, que pode compreender um número variável de simulações (configurável), e na posterior análise estatística, que permite gerar os gráficos dos valores médios das simulações. O tempo de simulação até à obtenção de resultados poderá variar também consoante o número de utilizadores definidos.

4.4.1. LIGAÇÃO APLICAÇÃO WEB – NS-2

Os valores definidos nas páginas de configuração e cenário são guardados em ficheiros de texto com extensão .txt. A interligação com o NS-2 é feita desta forma. O script de simulação .tcl lê esses ficheiros. De base está definido um ficheiro de simulação tcl. Através da leitura dos ficheiros gerados pela aplicação SimWimax, é possível incorporar no tcl base os valores definidos no SimWimax. Este processo é descrito pela seguinte figura:



Figura 33 Processo output SimWimax

O bloco Simulador NS-2 incorpora o ficheiro tcl utilizado para simular. De seguida apresenta-se, um exemplo da articulação entre o SimWimax e o simulador NS-2. A figura seguinte mostra um excerto da Página de Configuração Geral, nomeadamente a parte da selecção do codec VOIP.

Módulo VOIP

Seleccionar Codec

Codec Perfil A	<input type="text" value="G.711"/>	▼
Codec Perfil B	<input type="text" value="G.711"/>	▼
Codec Perfil C	<input type="text" value="G.711"/>	▼

Figura 34 Configuração codec na aplicação SimWimax

Ao ser clicado o botão “Gravar”, os parâmetros são submetidos por um formulário por intermédio do método GET:

```
<form id="form1" name="form1" method="get"
action="configdone.php">
```

Está definido como destino o ficheiro “configdone.php”. Nesta página são recebidos os parâmetros passados método GET e gravados nos ficheiros, neste caso no ficheiro config.txt, como descrito abaixo:

```
$codecvoip = $_GET['codecvoip'];
$fh = fopen($config, 'w') or die("can't open
file"); fwrite($fh, $codecvoip);
```


No exemplo anterior, na página `configdone.php` é recebida a variável `codecvoip` e é escrito no ficheiro `config.txt`.

No script de simulação no NS-2 esta variável é lida no ficheiro `config.txt`:

```
set config [open config.txt r]set codecvoip [gets  
$config]
```

Após a leitura do ficheiro `config.txt` é passado o valor lido a uma variável `codecvoip` no *script* tcl. Este é o processo de interligação entre o SimWimax e os *scripts* Otcl do NS-2.

Está assim concluído o processo de passagem de variáveis/parâmetros para o simulador através de ficheiros txt devidamente organizados. A leitura desses valores é feita de uma forma pré-estabelecida. Por exemplo, no ficheiro `config.txt` estão gravados todos os valores presentes na página de configuração. Para dar início à simulação, quando o utilizador clica no botão “Simular”, é utilizado um comando específico do PHP que interage com o sistema operativo, permitindo executar aplicações, scripts, etc. Neste caso foi feito um *script* em bash [35] que contém os comandos a serem executados.

Esse *script* de extensão. sh tem o seguinte formato:

```
#!/ bin/sh  
#executar o script tcl  
cd /home/jorgegilmoreira/wimax/ns-allinone-  
2.31/ns-2.31/  
cat /home/jorgegilmoreira/public_html/pass.txt |  
sudo -S ns sim3.tcl
```

Dentro deste ficheiro estão os comandos que são executados quando o *script* é chamado.

No PHP o comando utilizado é `exec`:

```
$executa = exec('sh  
home/jorgegilmoreira/wimax/ns-allinone-2.31/ns-  
2.31/sim3.sh', $retval4);
```

Após a execução do comando anterior, o simulador produz um ficheiro *trace* com toda a informação de actividade na rede definida. O SimWimax executa este *script* várias vezes para que sejam feitas simulações com sementes diferentes. Os ficheiros trace gerados em cada simulação são guardados separadamente, e tratados na fase seguinte: cálculo de taxas de transferência e atraso. Após estes cálculos, os respectivos gráficos são gerados dando origem à Página de Resultados.

4.4.2. RESULTADOS

O simulador NS-2 apenas tem como *output* um ficheiro *trace* que contém toda a actividade da rede. Desta forma, foi necessário desenvolver *scripts* para extrair a informação como taxa de transferência e atraso.

A linguagem utilizada para fazer o *parsing* do ficheiro Trace do NS-2 foi o AWK [36]. Esta linguagem é otimizada para tratar ficheiros organizados por coluna, é rápido e simples de programar. Os *scripts* utilizados foram desenvolvidos para o efeito tendo por base outros *scripts* disponíveis na internet [38] por outros utilizadores do simulador NS-2. Tendo estes *scripts* de base foi, no entanto, necessário efectuar as respectivas modificações para a adaptação dos mesmos ao cenário simulado.

A estrutura de um *script* AWK é bastante simples e é composta tipicamente por três partes. O início da *script*, onde são introduzidas as variáveis globais a ser utilizadas. De seguida, o corpo da *script*, que será executado para cada linha do *trace file* e, por fim, o terminar da *script* onde tipicamente serão impressos os resultados.

```
BEGIN {inicialização das variáveis}
{
  O conteúdo será executado a cada linha do trace
  file;
  $n -> retorna a variável presente na coluna n do
  trace file.
}
END{Imprime os dados}
```

Os extractos de código seguintes procuram demonstrar, de uma forma resumida, como este processo é realizado. Depois da inicialização de variáveis globais, no topo da *script*, o corpo da mesma inicia-se com a atribuição de variáveis às colunas do *trace file* necessárias ao cálculo da taxa de transferência e do atraso.

```
# Formato Trace: new wireless

event = $1
pkt_type=$5
time = $3
node_id = $5
flow_id = $39
pkt_id = $41
pkt_size = $37
```

No exemplo anterior foi definido o tipo de *trace* utilizado visto que a atribuição das variáveis às colunas difere quando se utiliza o novo tipo de trace do NS-2. Este foi um ponto a modificar em relação aos *scripts* convencionais. Quando se simula tecnologias sem

fios o *trace file* tem outro formato. É mais completo mas organizado de maneira diferente em relação ao *trace file* convencional.

De seguida, armazena-se os dados referentes aos tempos de envio de cada pacote.

```
# Guarda o tempo de envio dos pacotes
if (level == "AGT" && flow_id == flow && fonte
== src && event == "s") {
    (...)
    sendTime[pkt_id] = time
    (...)
}
```

No excerto de código anterior está criado um vector que vai armazenar todos os tempos de envio de pacotes no nível de aplicação com o identificador de fluxo igual (fid) ao que se pretende calcular e com o endereço de origem que se pretende obter.

De forma análoga são guardados os tempos de envio dos pacotes. É possível armazenar os tempos de chegada dos pacotes bem como a quantidade de *bits* que é recebida no destino.

```
# Actualiza o tamanho total de pacotes
recebidos e o seu tempo de chegada
if (level == "AGT" && flow_id == flow && node_id
== dst && event == "r" && fonte == src) {
    (...)
    # armazena o tamanho do pacote recebido
    recvdSize += pkt_size
    (...)
    # armazena o tempo de recepção do pacote
    recvTime[pkt_id] = time
    #printf("pacote recebido: %d no tempo
    (...)
```

No final do *script*, já é possível imprimir os valores da taxa de transferência instantânea.

```
END {
    (...)
    # Cálculo da taxa de transferência
    Tput = (recvdSize/(tempoFim-
tempoInicio))*(8/1000)

    printf ("%.2f, %.1f\n",tempoFim, Tput)
```

O *script* é executado pela aplicação SimWimax através do comando *exec* do PHP.

No *script* AWK o *parsing* é feito tendo como parâmetros, o fid, nó (ou SS) origem e destino. Assim deste modo são extraídos valores de débito (em Figura 35a) e atraso (em Figura 35b) independentes para cada flow.

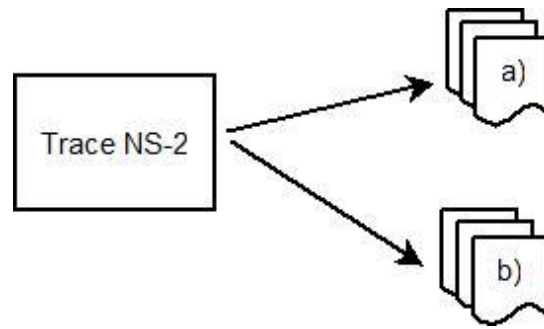


Figura 35 Parsing do ficheiro trace

Os *scripts*, um para débito e outro para o atraso, originam dois ficheiros em formato CSV (taxa de transferência e atraso) para cada *flow*. Um *flow* é um fluxo de dados por exemplo VOIP.

O ficheiro CSV contém, para o caso da taxa de transferência, duas colunas: uma com o tempo e outra com os valores instantâneos de débito. No ficheiro CSV do atraso existe uma coluna com o tempo e outra com o atraso instantâneo. Cada ficheiro CSV gerado está devidamente identificado pelo nome: no caso do cálculo de débito para *flow* 1 o nome atribuído é: debitoflow1.csv.

Todo o processo de geração de ficheiros csv é repetido em cada simulação. Todos os ficheiros são armazenados de forma estruturada. Do ponto de vista estrutural, está de acordo com a Figura 36:

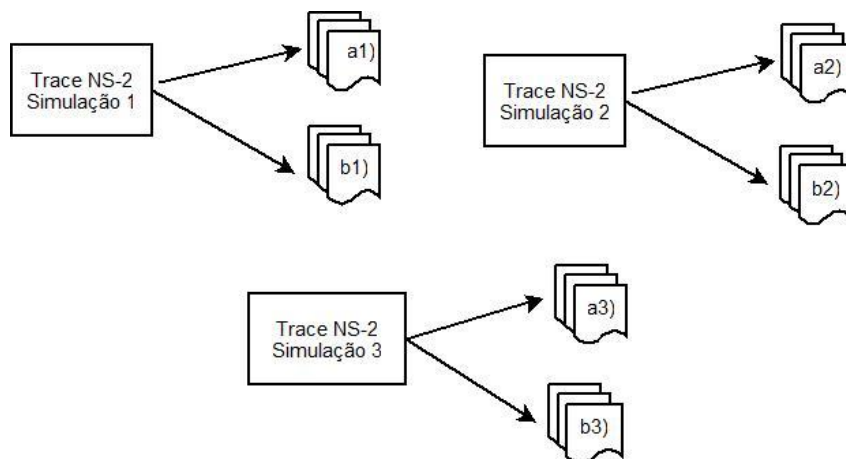


Figura 36 Parsing dos vários ficheiros trace

Na Figura 36 os ficheiros csv a1-a3 contêm valores de débito para cada *flow*. Os ficheiros csv b1 a b3 são ficheiros com valores de atraso.

Visto que os utilizadores são agrupados em perfis, o cálculo dos valores médios é feito por perfil. Assim teremos para cada perfil informação separada de média de débito e atraso para cada *flow* (Vod, VOIP, HTTP). Esta informação será apresentada na página resultados.

A cada ficheiro de Figura 36a1) a Figura 36a3) é feita a média dos valores de débito. Essa média é registada igualmente num ficheiro csv para tratamento posterior. O mesmo procedimento para os ficheiros Figura 36b1-b3 para o cálculo de valores médios de atraso.

No final são obtidos ficheiros CSV que representam médias para cada fluxo por perfil. Estes fluxos podem ser VoD, VOIP ou HTTP.

Cada habitação (cada SS) pode ter até três fluxos, um por cada tipo tráfego: Vod, VOIP ou HTTP. Os pesos dados a cada tráfego (nas percentagens) configurados na página do cenário podem, ou não, dar origem a um ou mais fluxos. Por exemplo, se o utilizador na percentagem de VOIP definir como 0% este tráfego não será gerado.

Apresentação de resultados

Com base nos ficheiros CSV médios por perfil para débito e atraso, são gerados gráficos.

O programa utilizado neste trabalho foi o Gnuplot [37] . O Gnuplot é uma ferramenta que permite gerar gráficos a partir da linha de comandos. Está disponível para grande parte dos sistemas operativos, Linux, Windows, Mac e Os/2. É disponibilizada gratuitamente em [37]. Foi criada originalmente para que estudantes e cientistas pudessem visualizar as fórmulas e funções desenvolvidas de uma forma interactiva. Recentes evoluções da ferramenta permitem programação não-iterativa para uso em *scripts web*.

O Gnuplot suporta uma grande variedade de tipos de *output*: terminal interactivo, *output* directo para impressora e mais importante para este trabalho o suporte para variados formatos como eps, jpeg, png, LaTeX, pdf, png entre outros.

A escolha do formato dos ficheiros em CSV vem ao encontro dos requisitos do Gnuplot. Um ficheiro organizado por colunas é o ideal para o Gnuplot.

Era um requisito deste trabalho obter uma ferramenta que gerasse uma imagem baseada num *script* também ele gerado automaticamente pela aplicação SimWimax. Assim, para

gerar um gráfico em formato imagem, basta construir o *script*. Quando este é chamado, o Gnuplot lê as linhas desse *script* e executa-as.

Exemplo da geração de um gráfico baseado num ficheiro CSV:

```
set terminal png truecolor
set autoscale
set style data lines
set output "testepplot2.png"
plot "testepplot.csv" using 1:2 title "tempo"
```

O ficheiro de configuração do Gnuplot tem a extensão. conf, no exemplo grafico.conf. Nas primeiras três linhas são opções do próprio Gnuplot. A quarta linha especifica o *output* que, neste caso, é uma imagem em formato png. Na última linha especifica-se o ficheiro de *input* para gerar o gráfico bem como o número das colunas que se quer utilizar no gráfico. O ficheiro CSV poderia ter mais colunas e, com o comando anterior, escolheria quais usar no gráfico.

Outras opções como atribuir nomes aos eixos x e y ou nomear o gráfico também são possíveis. Existem inúmeros tipos de gráficos e opções de configuração para eles.

No PHP este *script* é chamado utilizando o comando exec:

```
exec('sudo gnuplot grafico2.conf');
```

Cada ficheiro csv com média de débito ou atraso serve como *input* para o Gnuplot. São gerados gráficos para taxa de transferência e atraso médios para cada um dos tráfegos na página de resultados.

4.5. PROCESSO DE SIMULAÇÃO

A aplicação possui valores de configuração já previamente definidos. Exemplos disso são: a semente inicial, tempo de simulação, *codec* VOIP, carga do tráfego HTTP e sequência de vídeo para *streaming*.

A semente inicial é 1, tempo de simulação 100 segundos, *codec* VOIP é o G.711 e a sequência de vídeo é *Baseball with commercials*.

Ao aceder às Configurações Avançadas da aplicação é possível alterar os parâmetros QoS do sistema. São opções para utilização avançada.

Tipos de Resultados

Taxa de transferência

A taxa de transferência resulta da divisão entre o tamanho de um dado pacote e o tempo que este demora do nó emissor ao nó receptor ou vice-versa. Normalmente a taxa de transferência tem como unidades Kbit/s, Mbit/s ou Gbp/s.

Atraso

Considera-se atraso o tempo que um pacote demora desde a sua geração até ao seu destino numa rede. O pacote pode ser gerado na BS ou numa SS, dependendo se é *download* ou *upload*. É usual atribuir ao atraso a unidade de milissegundos (ms) ou segundos (s).

Jitter

O jitter é uma variação estatística do atraso, ou seja, pode ser definida como a medida de variação do atraso entre os pacotes sucessivos de dados. A unidade de medida desta grandeza é milissegundos (ms) ou segundos (s).

No entanto, até ao momento da escrita desta Tese não foi possível concluir as funcionalidades de determinação do jitter.

5. ANÁLISE DE RESULTADOS

Este capítulo reporta alguns dos testes realizados com o SimWimax. Esta aplicação mostra grande versatilidade de configuração e utilidade na obtenção de resultados.

Foram elaboradas várias simulações, com diferentes parametrizações de forma a demonstrar as potencialidades e funcionalidades da aplicação desenvolvida.

Todos os cenários consistem numa BS com várias SS uniformemente distribuídas. O comportamento de cada módulo foi avaliado em termos de taxa de transferência e atraso das ligações.

Na aplicação desenvolvida foram definidos alguns parâmetros fixos fazendo variar um só parâmetro de maneira a avaliar a influência dessa variação. Todas as simulações têm a duração de 100 segundos.

São apresentados 3 cenários de teste distintos com aspectos de configuração diferentes.

Cenário de teste 1

Este cenário compreende um número muito baixo de utilizadores. São utilizadores pertencentes ao mesmo perfil. Cada um dos utilizadores está configurado de acordo com os dados da Tabela 3. A figura seguinte mostra o Cenário proposto para teste.

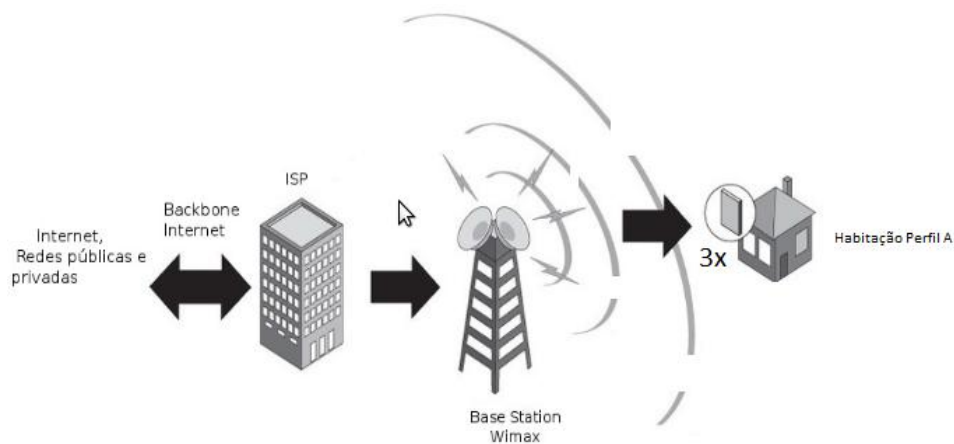


Figura 37 Cenário de teste 1

Neste cenário 1 foram definidos os seguintes parâmetros:

Habitações	Largura de Banda	VOIP	VoD	HTTP
3	10Mbit/s por habitação	20%. Codec: G.711	50%	Carga de 400kbit/s com distribuição HTTP-5

Tabela 3 Parâmetros Cenário Teste 1

Resultados obtidos:

Para o tráfego *Video on Demand* foram obtidos os gráficos presentes na Figura 38. De acordo com as características desta sequência de vídeo, *Baseball with Comercials* (Figura 17), o resultado está dentro do esperado. Taxa de transferência média de 348kbit/s obtida está muito acima do valor médio requerido que é de 144Kbit/s por serviços com estas características. O atraso médio é de 4.26ms, pelo que também, para este tipo de aplicação o valor obtido é bastante satisfatório em relação aos requisitos QoS.

A classe em que foi simulado é rtPS que, de acordo com os parâmetros QoS, admite um atraso máximo de 60ms.

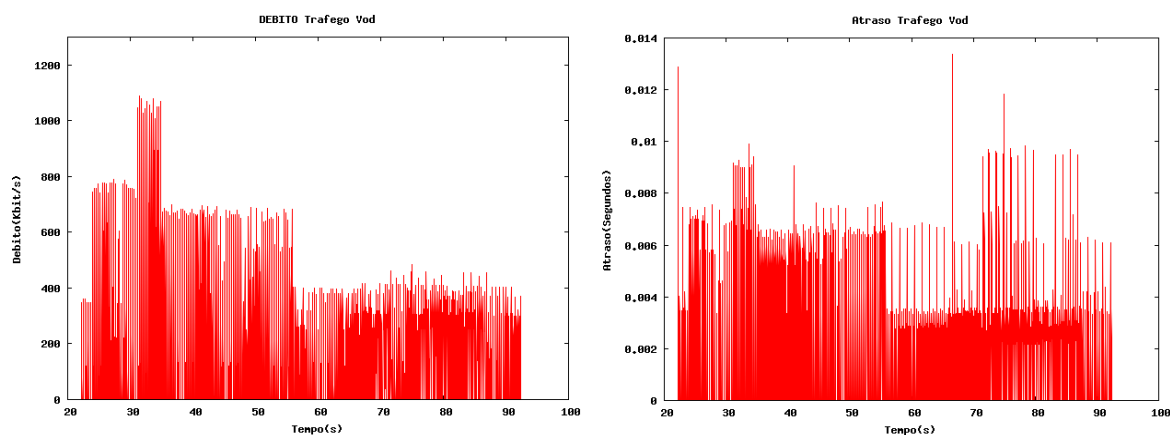


Figura 38 Taxa de transferência e atraso tráfego VoD Cenário 1

De seguida são apresentados na Figura 39 os resultados para o tráfego VOIP. A taxa de transferência média é de 32kbit/s e o atraso médio de 3.71ms. Visto que foi configurado um factor de utilização de 20% obteve-se um gráfico com menos pontos de utilização, ou seja, existem mais períodos de silêncio aproximadamente 80% do que os períodos em que existe conversação (20%).

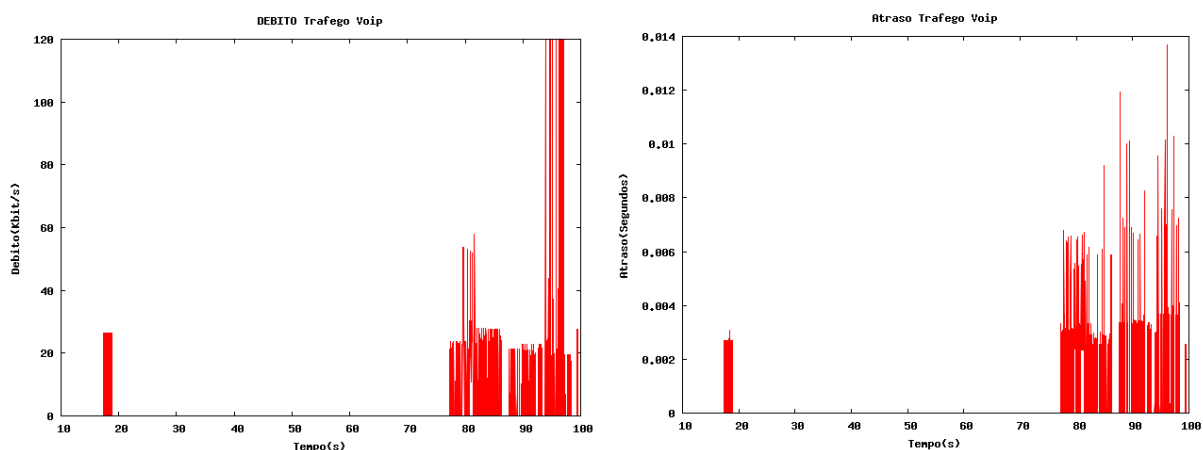
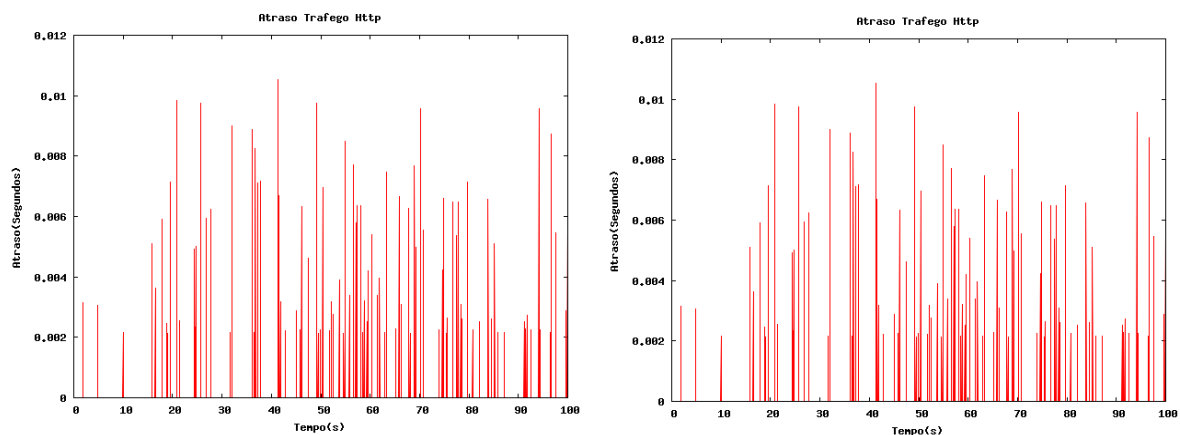


Figura 39 Taxa de transferência e atraso VOIP Cenário 1

Os gráficos obtidos para o tráfego HTTP são ilustrados na figura seguinte. A taxa de transferência média é de 428kbit/s, respeitando o que foi parametrizado. O atraso de 4.28ms está dentro do valor requerido para a classe de serviço configurada.



Com a distribuição utilizada HTTP-5 constata-se que o módulo gera pacotes segundo a segundo com diferentes tamanhos, motivo para a variação da taxa de transferência.

Tendo em conta a configuração utilizada para este teste, conclui-se que pelos valores médios obtidos os serviços funcionam correctamente.

Cenário de teste 1.1

Este cenário é muito semelhante ao primeiro, excepto na distribuição escolhida para o tráfego HTTP.

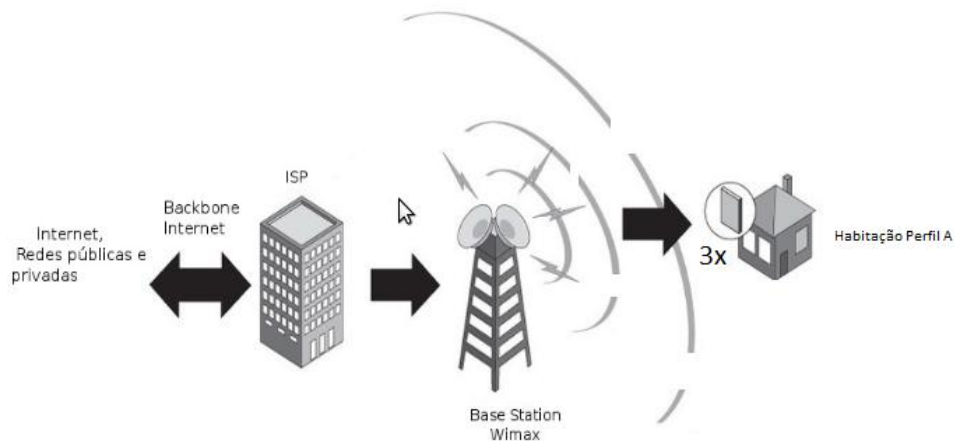


Figura 40 Cenário para teste 1.1

A figura acima demonstra o cenário de teste 1.1.

A tabela seguinte mostra a configuração para este teste.

Habitacões	Largura de Banda	VOIP	VoD	Http
3	10Mbit/s por habitação	20%. Codec: G.711	50%	Carga de 400kbit/s com distribuição HTTP-1

Tabela 4 Parâmetros para Cenário teste 1.1

Irá ser observado o efeito da mudança da distribuição. A figura seguinte contém os gráficos de taxa de transferência e atraso para o tráfego HTTP com distribuição HTTP-1.

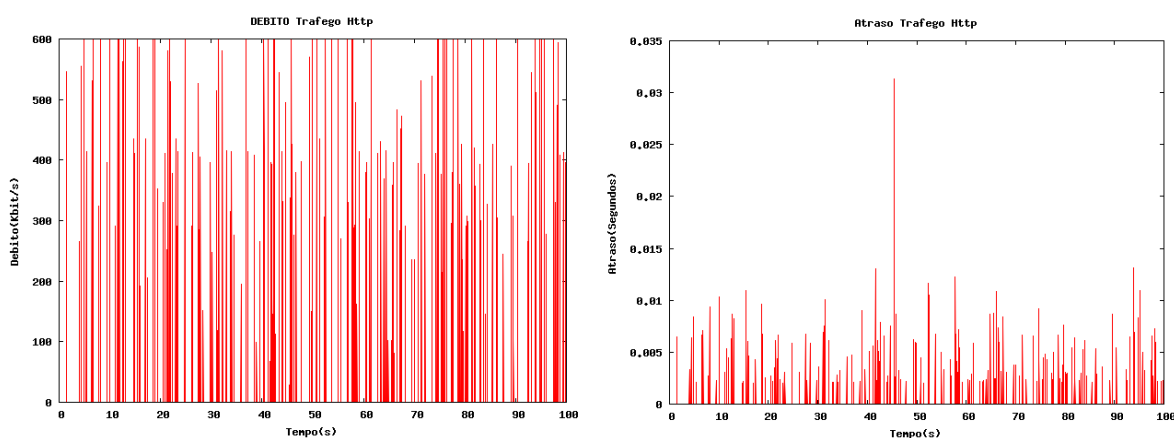


Figura 41 Tráfego HTTP com distribuição HTTP-1

Constata-se que a distribuição escolhida, consegue caracterizar um outro tipo de cliente, mais activo no que respeita aos serviços HTTP. O maior número de amostras no gráfico mostra isso mesmo. A taxa de transferência do HTTP subiu um pouco para os 455Kbit/s. O atraso desceu ligeiramente para 3.9ms.

Esta alteração não afectou o resto do sistema, ocorreram apenas variações muito pequenas, inferiores a 2%.

Com a presente configuração consegue-se garantir que todos os serviços funcionam correctamente. Todos os utilizadores estarão satisfeitos com os serviços contratados.

Cenário de teste 1.3

O próximo cenário de teste pode ser descrito pela figura seguinte:

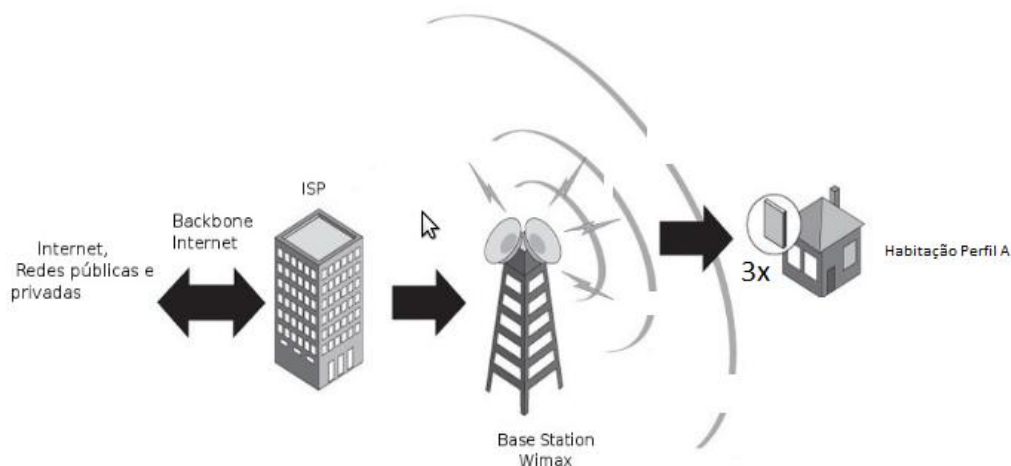


Figura 42 Cenário de teste 1.3

A configuração para este teste é o seguinte:

Habitações	Largura de Banda	VOIP	VoD	HTTP
3	10Mbit/s por cliente	20%. Codec: GSM.AMR	50%	Carga de 400kbit/s com distribuição HTTP-1

Tabela 5 Parâmetros para Cenário de teste 1.3

Resultados obtidos:

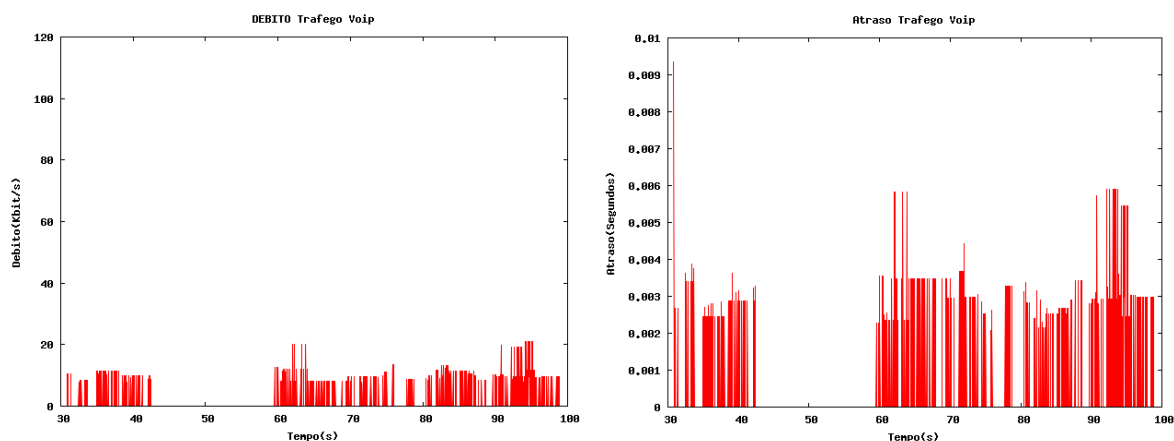


Figura 43 Taxa de transferência e Atraso VOIP cenário 1.3

Valores de 3.1ms para o atraso médio e taxa de transferência média de 11.1kbit/s. Esta configuração é útil para os casos em que se queira retratar um tipo de cliente que contrata um serviço telefónico VOIP de baixo custo.

Os autores do módulo VOIP mencionam este *codec* como o que tem menor qualidade de voz. Os valores encontrados vão de encontro ao que foi configurado.

Existe diminuição da taxa de transferência o que reflecte a escolha do um *codec* de inferior qualidade de voz. Os valores de atraso são aceitáveis para este tipo de serviço VOIP.

Os restantes tráfegos não sofreram alterações. Com a actual configuração garante-se boa qualidade nos serviços, visto que estes funcionam como esperado.

Cenário 2

Neste cenário de teste, vai ser avaliada a resposta do sistema ao aumento de utilizadores como mostra a figura seguinte:

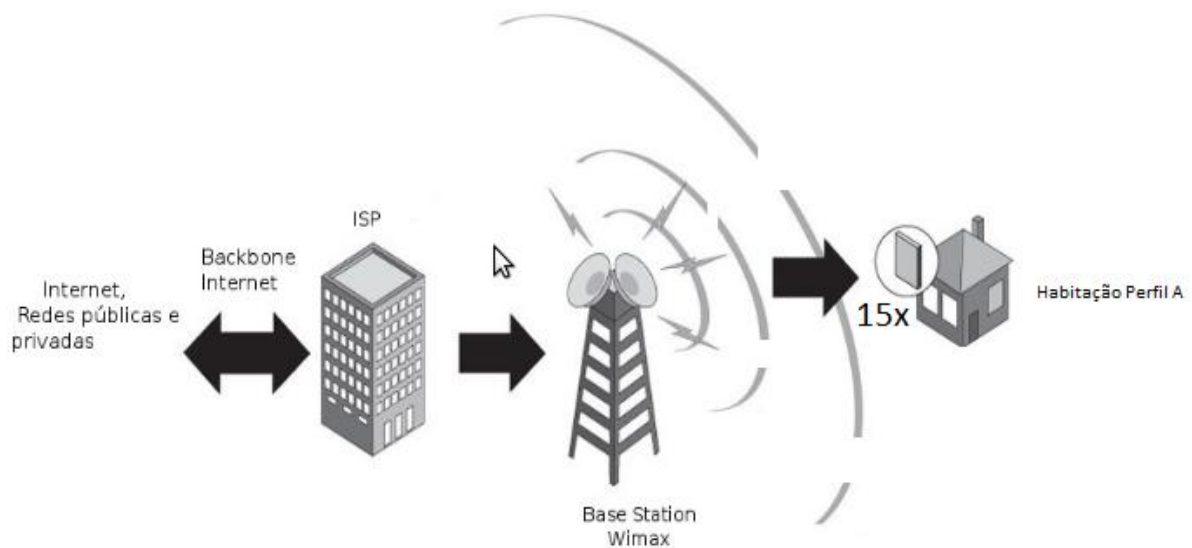


Figura 44 Cenário de teste 2

Os detalhes da simulação estão na tabela seguinte:

Habitações	Largura de Banda	VOIP	VoD	HTTP
15	10Mbit/s por habitação	20%. Codec: G.711	50%	Carga de 400kbit/s com distribuição HTTP-1

Tabela 6 Parâmetros para Cenário de teste 2

Os gráficos para o tráfego *Video-on-demand* na figura seguinte:

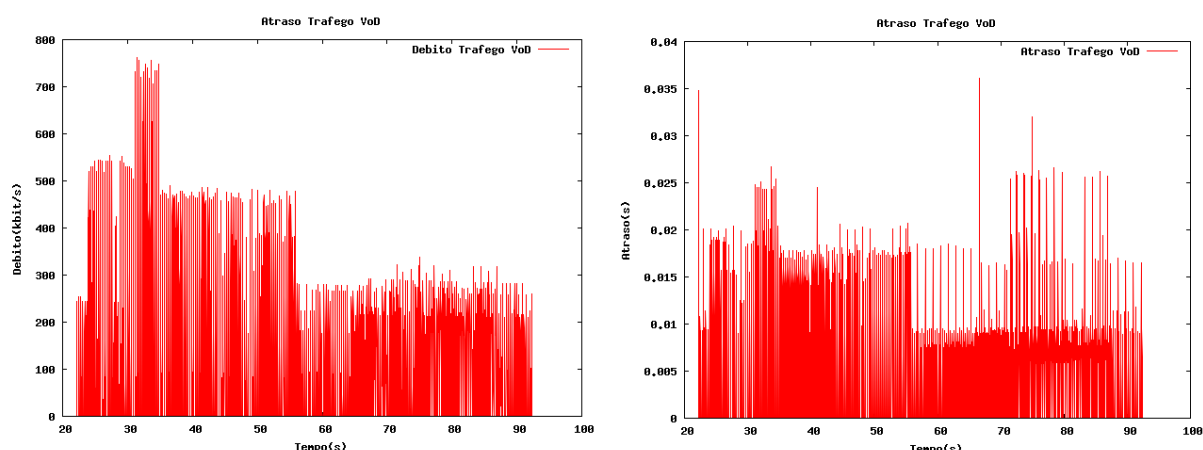


Figura 45 Taxa de transferência VoD Cenário 2

Na figura anterior regista-se um aumento de atraso para 14.2ms. A taxa de transferência desce para 280kbit/s, valor médio.

Na figura seguinte é mostrada a taxa de transferência e atraso para o tráfego VOIP:

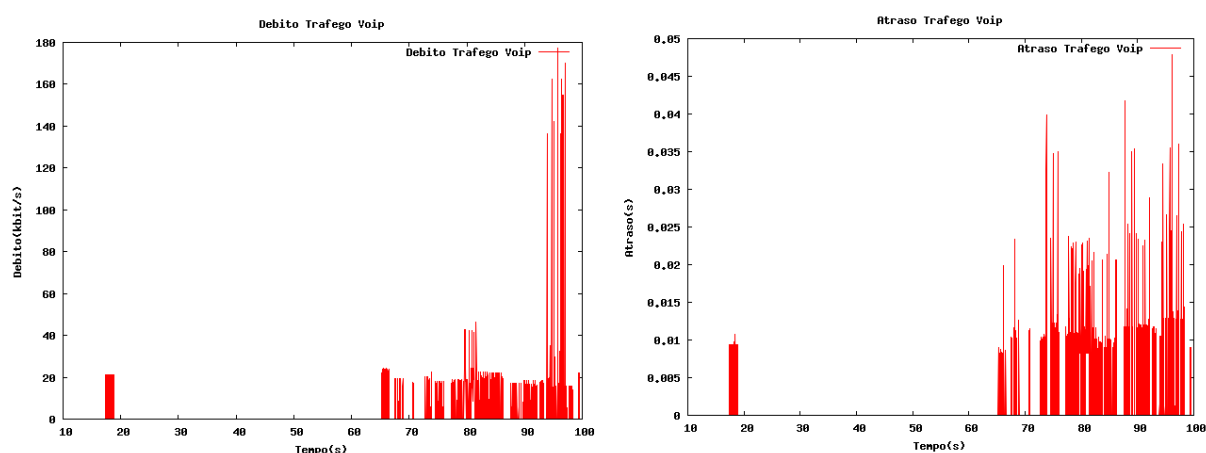


Figura 46 Taxa de transferência VOIP Cenário 2

Na próxima figura a taxa de transferência e atraso médios para tráfego HTTP.

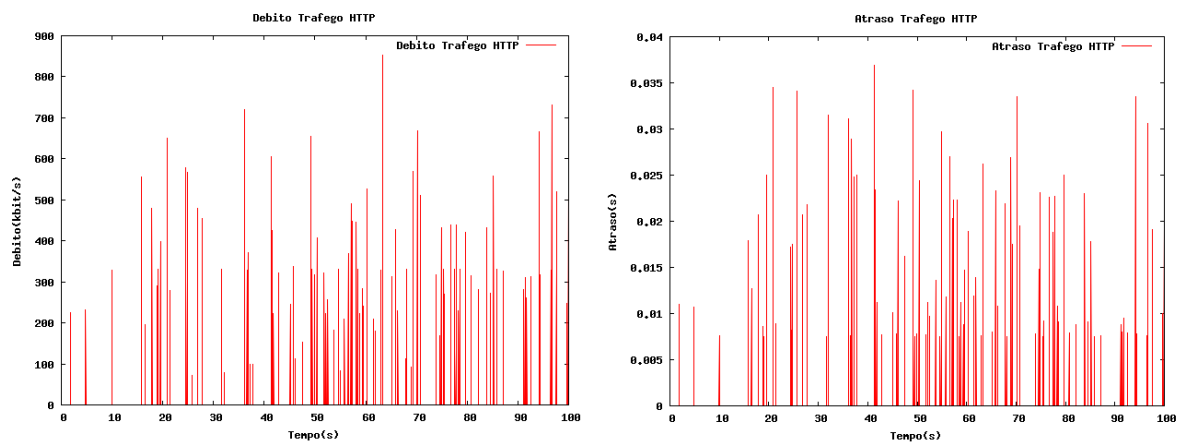


Figura 47 Taxa de transferência e atraso HTTP Cenário 2

Tanto no VOIP como no HTTP registou-se, mais uma vez, aumentos de atraso e diminuição da taxa de transferência média.

Cenário de teste 3

Relativamente a este cenário de teste, apenas se vai aumentar o número de utilizadores como mostra a figura seguinte:

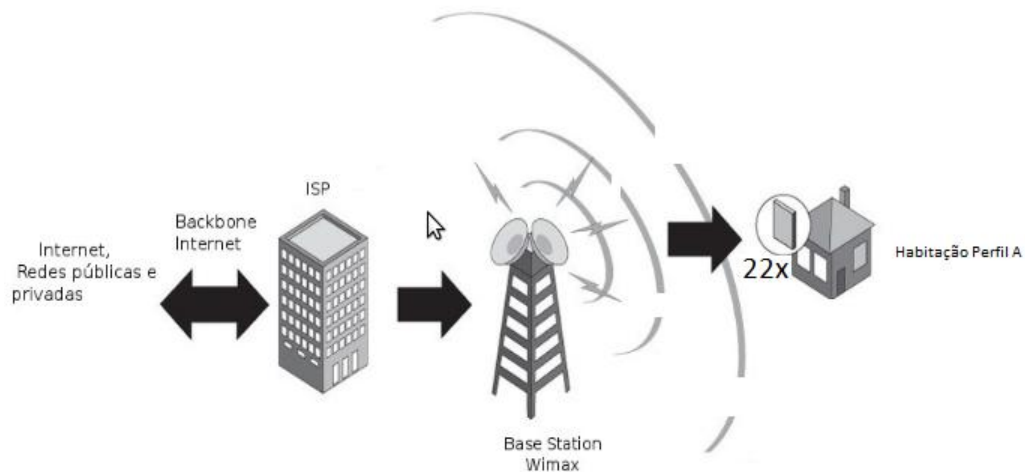


Figura 48 Cenário de teste 3

A tabela seguinte mostra os detalhes da simulação:

Habitações	Largura de Banda	VOIP	VoD	HTTP
22	10Mbit/s por habitação	20%. Codec: G.711	50%	Carga de 400kbit/s com distribuição HTTP-1

Tabela 7 Parâmetros Cenário de teste 3

De seguida, os resultados para o tráfego VoD são ilustrados na Figura 49.

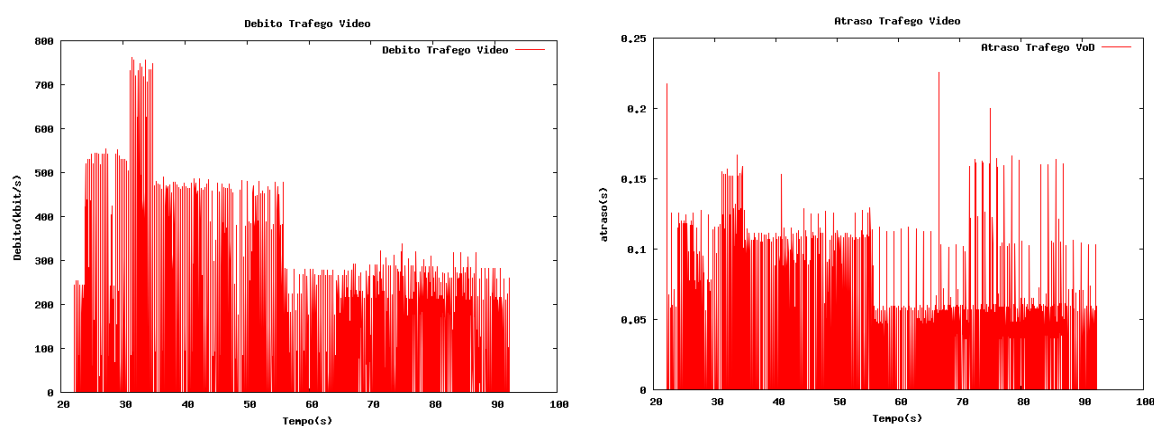


Figura 49 Taxa de transferência e atraso VoD Cenário 3

Os atrasos são muito altos impossibilitando o utilizador de visualizar a sequência de vídeo. Atraso por vezes superior a 100ms.

Na figura seguinte pode-se constatar que a taxa média de transferência e atraso, mais uma vez, fora dos parâmetros de maneira a garantir uma chamada de voz com qualidade.

A taxa de transferência está muito abaixo do configurado (Codec G.711) e com um atraso excessivo.

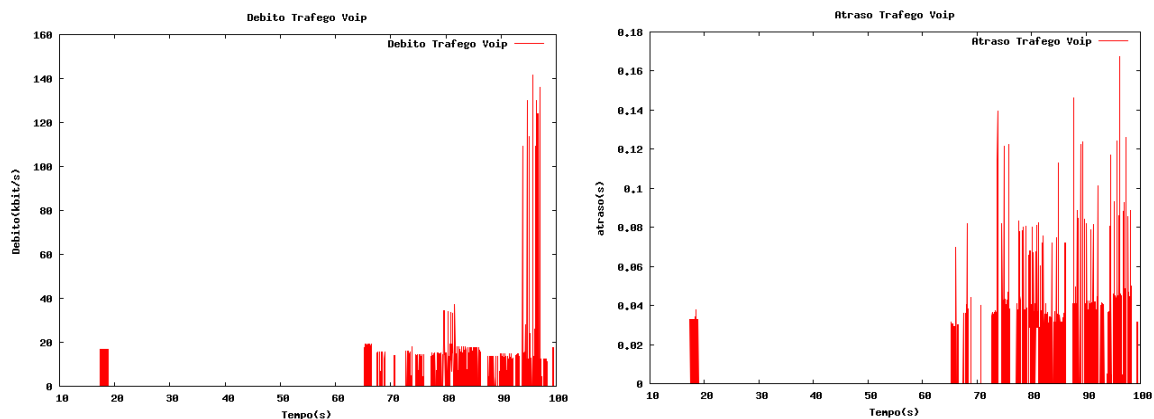


Figura 50 Taxa de transferência VoD Cenário 3

O mesmo se passa com o tráfego HTTP na figura seguinte. Como se pode observar na figura seguinte, os atrasos são elevados. Numa situação real, o cliente/utilizador não teria uma navegação na internet fluída.

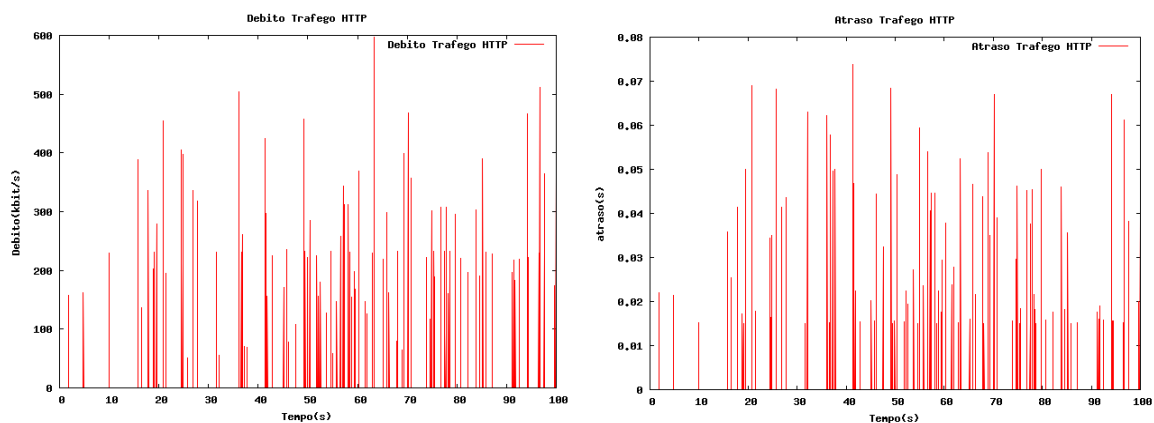


Figura 51 Taxa de transferência e atraso HTTP Cenário 3

Com base nos resultados obtidos, pode-se aferir que o serviço fornecido aos clientes está abaixo dos níveis mínimos.

6. CONCLUSÕES

O estudo de cenários residências é complexo, visto que o universo de utilizadores em cada situação pode variar. O SimWimax possui um conjunto de opções que permitem a caracterização de grupos de utilizadores por tipo de serviço VOIP, VoD ou HTTP. Na aplicação, é possível agrupar os clientes em perfis e, para cada um, definir factores de utilização dos serviços.

É possível definir a dimensão de cada cenário bem como um conjunto de opções para cada modelo de tráfego o que dá ao projectista bastante flexibilidade no uso da ferramenta.

No final, o utilizador do SimWimax obtém uma série de resultados sob a forma de gráficos para mais facilmente avaliar o cenário simulado.

A aplicação desenvolvida permite realizar de uma forma mais rápida a simulação e parametrização de cenários de utilização da tecnologia WIMAX para servir conjuntos de utilizadores.

A elaboração do SimWimax requereu estudo em detalhe do simulador NS-2 que tomou parte do projecto visto ser uma ferramenta de simulação complexa e pouco intuitiva, apesar de muito versátil e poderosa. Outra parte do trabalho bastante morosa foi a elaboração dos

scripts e toda a automatização e interacção com o simulador na qual assenta a aplicação WIMAX.

A aplicação SimWimax mostrou ser de enorme utilidade no auxílio de projectos de dimensionamento de ligações WIMAX para grupos de utilizadores. Qualquer projectista ou estudante/investigador que queira simular este tipo de cenários, fá-lo de uma maneira simples, sem ter conhecimento algum do simulador NS-2.

Dado que o sistema SimWimax faz todo o processo de simulação e tratamento de dados, cada simulação despende de elevados recursos computacionais. Por este facto, o SimWimax não é um simulador rápido no computador que foi usado para o seu desenvolvimento. Esse tipo de configurações mais exigentes no que respeita ao processamento, devem ser simuladas em computadores com recursos disponíveis para o efeito, nomeadamente, com maior capacidade de processamento e de memória.

6.1. DESENVOLVIMENTOS FUTUROS

A aplicação SimWimax para além das opções que tem disponível, pode ser evoluída com a adição de um maior número de cenários, tecnologias de acesso e/ou modelos de tráfego.

Neste trabalho, não foi possível descrever com detalhe todos os aspectos de implementação do software. Deste modo, seria interessante disponibilizar um documento para que seja possível a continuação do desenvolvimento desta ferramenta.

Referências Documentais

- [1] IEEE. “IEEE Standard for Local and metropolitan area networks Part 16: Air Interface for Fixed Broadband Wireless Access Systems”, IEEE Standard, 802.16 , 2004.
- [2] Mobile WIMAX / Edited by Kwang-Cheng Chen, J Roberto B. de Marca p. cm ISBN 978-0-470-51941-7.
- [3] IEEE Std 802.16-2004 – Part 16 – Air Interface for Fixed Broadband Wireless Access Systems. Technical report, June 2004. IEEE Standard for Local and metropolitan area networks.Fundamentals of WIMAX : understanding broadband wireless networking / Jeffrey G. Andrews, Arunabha Ghosh, Rias
- [4] WIMAX: Technology for Broadband Wireless Access ISBN: 978-0-470-02808-7
- [5] NAIR Govindan; CHOU Joey; MADEJSKI Tomasz; PERYCZ Krzysztof; PUTZOLU David; SYDIR Jerry – IEEE 802.16 Medium Access Control and Service Provisioning. Intel® Technology Journal, Volume 8, Issue 03, pag.213-228, 2004.
- [6] Fabio Magagnini Quinelato & Antônio Marcos Alberti : ANÁLISE DAS CARACTERÍSTICAS DE FUNCIONAMENTO DA GERÊNCIA DE BANDA EM REDES WIMAX
- [7] The Network Simulator – NS-2: www.isi.edu/nsnam/ns/.
- [8] Fernando Miguel Ribeiro Bartilotti de Matos: QoS em WIMAX – Análise e avaliação de módulos para simulação com o ns-2.
- [9] NIST – The Network Simulator NS-2 NIST add-on – IEEE 802.16 model (MAC + PHY). 2007.
- [10] NDSL; CHENG Jenhui – The Design & Implementation of WIMAX Module for ns-2 Simulator.
- [11] NDSL – WIMAX for NS-2. http://ndsl.csie.cgu.edu.tw/wimax_ns2.php.
- [12] FREITAG Juliana; FONSECA Nelson – Um módulo para simulação de redes WIMAX no simulador ns-2. 2008.
- [13] UNICAMP – WIMAX module for the ns-2 simulator. http://www.lrc.ic.unicamp.br/wimax_ns2/ .
- [14] Voip for NS-2 Simulator: <http://info.iet.unipi.it/~cng/ns2voip/>.
- [15] User-level Performance Evaluation of VoIP Using ns-2: A. Bacioccola, C. Cicconetti, G. Stea Dipartimento di Ingegneria dell’Informazione University of Pisa, Via Diotisalvi, 2 – 56100 Pisa, ITALY
- [16] A.P. Markopoulou, F. A. Tobagi, and M. J. Karam. Assessment of VoIP quality over Internet backbones. Proc. IEEE INFOCOM 2002, New York, USA, June 23–27, pp. 150–159.

- [17] Pierre L'Ecuyer. Software for uniform random number generation: Distinguishing the good and the bad. In Proceedings of the 2001 Winter Simulation Conference, pages 95–105, December 2001.
- [18] MPEG-4 and H.263 Video Traces for Network Performance Evaluation: <http://www.tkn.tu-berlin.de/research/trace/trace.html>
- [19] MPEG-4 and H.263 Video Traces for Network Performance Evaluation: Frank H.P.Fitzek, Martin Reisslein
- [20] Software BTTVGRAB <http://freshmeat.net/projects/bttvgrab/>
- [21] O.Rose. Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems. Technical Report 101, University of Wuerzburg, Institute of Computer Science, Am Hubland, 97074 Wuerzburg, Germany, February 1995.
- [22] Modelagem de Tráfego HTTP: Desenvolvimento e Aplicação: Kleber Vieira Cardoso. Rio de Janeiro 2002.
- [23] Mah, B. An empirical model of http network traffic. In Proc. INFOCOM'97 (Abril 1997).
- [24] ABRAHAMSON, H. AND AHLGREN, B. Using Empirical Distributions to Characterize Web Client Traffic and to Generate Synthetic Traffic. In IEEE/Globecom'00 (San Francisco, Novembro de 2000)
- [25] CHOI, H.-K. AND LIMB. J. O. A Behavioural Model of Web Traffic. In Internacional Conference of Networking Protocol 99 (ICNP99) (1999)
- [26] MOLINA, M., CASTELLI, P. AND FODDIS, G Web Traffic Modeling Exploting TCP Connections Temporal Clustering through HTML-REDUCE. IEEE Network Magazine 14, 3 (2000), 46-55.
- [27] BARFORD, P, AND CROVELLA, M. Generating representative web workloads for network and server performance evaluation. In Proc. ACM SIGMETRICS Conference (Julho 1998), pp.151-160.
- [28] PARK, K., KIM, G., AND CROVELLA, M On the relationship between file sizes, transport protocols, and self-similar network traffic. In Proc. IEEE Internacional Conference on Network Protocols (Outubro de 1996), pp. 171-180.
- [29] BARDFORD, P. BESTAVROS, A. BRADLEY, A., AND CROVELLA, M.Changes in Web Client Access patterns: characteristics and Caching Implications. In Special Issue on Characterization and Performance Evaluation (1999).
- [30] REYES-LECUONA, A.GONZALEZ-PARADA, E. CASILARI, E. AND DIAZ-ESTRELLA, A. Apage-oriented WWW traffic model for wireless system simulations. In Proc. Of the 16th Internacional Teletraffic Congress (ITC'16) (Julho 1999).
- [31] CROVELLA, M.E. AND BESTAVROS A Self-similarity in WWW traffic: Evidence and possible causes. In Proc. Of the ACM SIGMETRICS Conference on Measurement & Modeling of Computer Systems (Philadelphia 1996), pp.160-199

- [32] VOGUE: The Momusys Video Object Generator with User Enviroment:
B.Marcotegui, P.Correia, F.Marqués, R.Mech, R.Rosa, M.Wollborn and F.Zanoguera
- [33] PHP: Hypertext Preprocessor: <http://www.php.net>
- [34] XAMPP FOR LINUX: <http://www.apachefriends.org/en/xampp-linux.html>
- [35] BASH: <http://www.gnu.org/software/bash/#introduction>
- [36] AWK: <http://www.gnu.org/manual/gawk/gawk.html>
- [37] GNUPLOT: <http://www.gnuplot.info/>
- [38] Scripts for NS-2 Marco Foire <http://perso.citi.insa-lyon.fr/mfiore/>
- [39] NS by Example <http://nile.wpi.edu/NS/>
- [40] Rafael CASTRO-PENA, “WIMAX Networks”, On-Line: <http://www.supinfo-projects.com/cn/>, Paris Academy of Computer Science, SUPINFO Projects, Ecole Sup’erieure D’Informatique, Mai. 2005
- [41] WIMAX Industry: <http://www.wimax-industry.com>